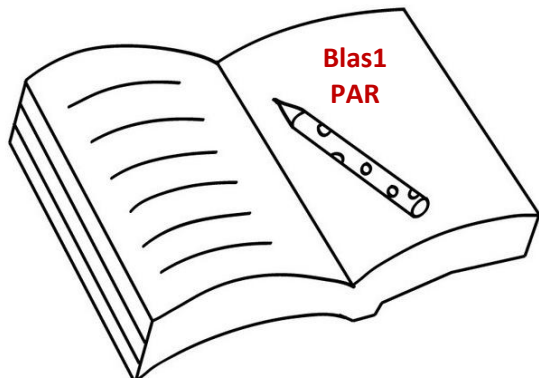
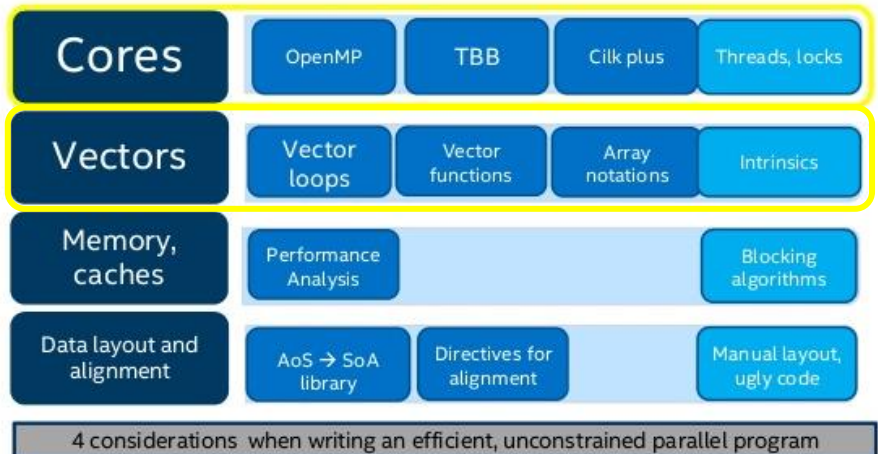


Parallel Programming for Intel® Architecture (or, in general, for normal CPUs)



Los programas que a continuación se proponen, es un subconjunto del Basic Linear Algebra Subprogramas I. Su resolución se ha realizado con distintas versiones del sistema operativo, compilador gcc y en distintas máquinas.

1.- Intel i7-4810MQ 2.8 GHz

4 cores, HT, AVX2, FMA, Turbo Boost.

Cache Datos: L1 4x32KB L2 4x256KB L3 6MB.

Gcc 4.8.1

Windows 7. Opción Energía: Equilibrado



2.- Intel Core i5 4200H 2.8GHz

2 cores, HT, AVX2, FMA, Turbo Boost.

Cache Datos: L1 2x32KB L2 2x256KB L3 3MB.

Gcc 4.8.1

Windows 10. Opción Energía: Equilibrado



3.- Intel Core Quad CPU Q9550 2.83Hz

4 cores, sse4.1

Cache Datos: L1 4x32KB L2 2x6MB

Gcc 4.4.0

Windows 7. Opción Energía: Equilibrado



Indice.

Copy	5
icopy	5
scopy.....	11
dcopy	15
Scal	20
iscal.....	20
sscal.....	25
dscal	29
dot.....	33
idot	33
sdot.....	40
ddot	47
dsdot	51
sdsdot	55
amax	59
zi-amax	59
si-amax.....	64
di-amax	68
norm2	71
snorm2	71
dnorm2.....	76

asum	79
iasum.....	79
sasum	82
dasum	85
swap	88
iswap.....	88
sswap	91
dswap.....	94
axpy	97
iaxpy	97
saxpy	100
daxpy	103
rot	106
irrot	106
srot	109
drot.....	112
rotm	115
irrotm.....	115
srotm	119
drotm	122

Copy

icopy

```

3 void icopy_seq(int n, int *x, int *y)
4 {
5     int i;
6     for (i = 0; i < n; ++i)
7         y[i] = x[i];
8 }

```

```

10 static void align_icopy_sse(int n, int *x, int *y, int nprol, int nvecl)
11 {
12     int i;
13     __m128i v_x, v_a;
14
15     //Prologo
16     for(i = 0; i < nprol ; i++)
17         y[i] = x[i];
18
19     //Cuerpo
20     for (i = nprol; i < nvecl; i+=4){
21         v_x = _mm_load_si128((__m128i *)&x[i]);
22         _mm_store_si128((__m128i *)&y[i],v_x);
23     }
24
25     //Epilogo
26     for(i = nvecl; i < n ; i++)
27         y[i] = x[i];
28 }
29
30 static void noalign_icopy_sse(int n, int *x, int *y, int nprol, int nvecl)
31 {
32     int i;
33     __m128i v_x, v_a;
34
35     //Prologo
36     for(i = 0; i < nprol ; i++)
37         y[i] = x[i];
38
39     //Cuerpo
40     for (i = nprol; i < nvecl; i+=4){
41         v_x = _mm_load_si128((__m128i *)&x[i]);
42         _mm_storeu_si128((__m128i *)&y[i],v_x);
43     }
44
45     //Epilogo
46     for(i = nvecl; i < n ; i++)
47         y[i] = x[i];
48 }
49 static void icopy_sse(int n, int *x, int *y)
50 {
51
52     int dir1 = (int) (&x[0]);
53     int nprol1 = 16 - ((dir1>>2)%16);
54     int nvecl = (((int) floor((double)((n-nprol1)/4.0)))<<2) + nprol1;
55
56     int dir2 = (int) (&y[0]);
57     int nprol2 = 16 - ((dir2>>2)%16);
58
59     if(nprol1 == 16) nprol1 = 0;
60     if(nprol2 == 16) nprol2 = 0;
61
62     if(nprol1 == nprol2)
63         align_icopy_sse(n,x,y, nprol1, nvecl);
64     else
65         noalign_icopy_sse(n,x,y, nprol1, nvecl);
66 }

```

```

1 #include <xmmintrin.h> //SSE
2 #include <emmintrin.h> //SSE2
3 #include <pmmmintrin.h> //SSE3
4 #include <smmmintrin.h> //SSE4.1

```

```

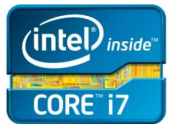
6 static void align_icopy_avx(int n, int *x, int *y, int nprol, int nvecl)
7 {
8     int i;
9     → __m256i v_x, v_y, v_a;
10
11     //Prologo
12     for(i = 0; i < nprol ; i++)
13         y[i] = x[i];
14
15     //Cuerpo
16     for (i = nprol; i < nvecl; i+=8){
17     → v_x = _mm256_load_si256((__m256i *)&x[i]);;
18     → _mm256_store_si256((__m256i *)&y[i],v_x);
19     }
20
21     //Epilogo
22     for(i = nvecl; i < n ; i++)
23         y[i] = x[i];
24 }
25
26 static void noalign_icopy_avx(int n, int *x, int *y, int nprol, int nvecl)
27 {
28     int i;
29     __m256i v_x, v_y, v_a;
30
31     //Prologo
32     for(i = 0; i < nprol ; i++)
33         y[i] = x[i];
34
35     //Cuerpo
36     for (i = nprol; i < nvecl; i+=8){
37         v_x = _mm256_load_si256((__m256i *)&x[i]);;
38     → _mm256_storeu_si256((__m256i *)&y[i],v_x);
39     }
40
41     //Epilogo
42     for(i = nvecl; i < n ; i++)
43         y[i] = x[i];
44 }
45 static void icopy_avx(int n, int *x, int *y)
46 {
47
48     int dir1 = (int) (&x[0]);
49     int nprol1 = 16 - ((dir1>>2)%16);
50     int nvecl = (((int) floor(((double) ((n-nprol1)/8.0)))<<3)+ nprol1);
51
52     int dir2 = (int) (&y[0]);
53     int nprol2 = 16 - ((dir2>>2)%16);
54
55     if(nprol1 == 16) nprol1 = 0;
56     if(nprol2 == 16) nprol2 = 0;
57
58     if(nprol1 == nprol2)
59         align_icopy_avx(n,x,y, nprol1, nvecl);
60     else
61         noalign_icopy_avx(n,x,y, nprol1, nvecl);
62 }

```

```
1  #include <omp.h>

4  void icopy_omp(int n, int *x, int *y)
5  {
6      int i;
7      ➔ #pragma omp parallel for
8         for (i = 0; i < n; ++i)
9             y[i] = x[i];
10
11 }
```

```
5  #define min(x,y) ((x)<(y)) ? (x) : (y)
6
7  #include "icopy_sse.c"
8  #include "icopy_avx.c"
9
10 static void icopy_omp_sse(int n, int *x, int *y)
11 {
12     int i,trozo,ini,fin;
13
14     trozo=ceil((float)n/(float)NTHR);
15
16     ➔ #pragma omp parallel private(i,ini,fin) firstprivate (n,trozo)
17     {
18         i = omp_get_thread_num();
19         ini=i*trozo;
20         fin=min(trozo*(i+1),n);
21         ➔ icopy_sse(fin-ini,&x[ini],&y[ini]);
22     }
23 }
24
25 static void icopy_omp_avx(int n, int *x, int *y)
26 {
27     int i,trozo,ini,fin;
28
29     trozo=ceil((float)n/(float)NTHR);
30
31     ➔ #pragma omp parallel private(i,ini,fin) firstprivate (n,trozo)
32     {
33         i = omp_get_thread_num();
34         ini=i*trozo;
35         fin=min(trozo*(i+1),n);
36         ➔ icopy_avx(fin-ini,&x[ini],&y[ini]);
37     }
38 }
```



TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	3.068681	2.978667	0.003955	0.005259	0.005377
4	1000	3.555195	3.581357	0.008963	0.008720	0.001946
4	1500	3.796460	3.683457	0.012656	0.010797	0.011255
4	2000	4.027294	4.081068	0.016756	0.002049	0.009939
4	2500	4.219146	4.261286	0.024171	0.022804	0.025490
4	3000	4.246184	4.183720	0.024094	0.022537	0.030898
4	3500	3.910071	3.581407	0.029062	0.027253	0.023671
4	4000	3.262078	3.329094	0.019849	0.018465	0.038343
4	8000	1.775762	1.908893	0.072558	0.073325	0.083954
4	13000	2.068732	2.042320	0.106085	0.093109	0.106765
4	18000	2.150510	2.153798	0.143382	0.138346	0.139820
4	23000	2.195744	1.884203	0.182260	0.185354	0.174334
4	28000	2.218237	2.103216	0.241260	0.230934	0.210606
4	33000	2.207596	2.129317	0.268835	0.241690	0.218508
4	38000	1.933296	1.879598	0.330683	0.338530	0.295625
4	43000	1.706912	1.726851	0.397786	0.342156	0.401750
4	48000	1.625907	1.606697	0.354687	0.394111	0.359347
4	53000	1.600219	1.615955	0.427159	0.402807	0.431556
4	58000	1.605534	1.617607	0.500037	0.439188	0.498193
4	100000	1.635841	1.644452	0.703744	0.730898	0.769668
4	175000	1.596360	1.600914	1.005191	1.178831	1.194235
4	250000	1.549469	1.568976	1.085412	1.438569	1.446782
4	325000	1.580373	1.570536	1.214633	1.507857	1.589479
4	400000	1.548285	1.541028	1.433238	1.960450	1.637683
4	475000	2.173760	2.167386	1.902923	2.541180	2.571452
4	550000	1.531451	1.593352	1.413036	1.590372	1.759294
4	625000	1.490839	1.567962	1.538267	2.167560	2.541163
4	700000	1.432445	1.118873	1.318519	2.167181	2.048183
4	775000	1.525396	1.528111	1.827949	2.160909	1.795944
4	1600000	1.328215	1.275412	1.741654	2.023923	2.055568
8	250000	1.597153	1.542670	1.134089	1.323084	1.476284
8	325000	1.555928	1.553008	1.359027	1.747567	1.641883
8	400000	1.602966	1.626699	1.606387	1.757136	1.815787
8	475000	1.543618	1.566500	1.583831	1.893418	1.821046
8	550000	1.569188	1.578385	1.670609	1.926579	2.340924
8	625000	1.535213	1.404066	1.245431	1.757320	1.776029
8	700000	1.504948	1.513630	1.927220	2.362661	2.087590
8	775000	1.485955	1.504818	1.830789	2.399113	2.511812
8	1600000	1.187150	1.489911	2.375558	2.466479	2.707735

	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	3,170929	3,236264	0,006558	0,007286	0,007323
	2	1000	3,472381	3,555339	0,013179	0,017095	0,017846
	2	1500	3,814464	3,79646	0,010947	0,013152	0,01293
	2	2000	4,966557	4,492544	0,016454	0,027813	0,030972
	2	2500	3,617104	3,759596	0,046964	0,034867	0,042779
	2	3000	3,450619	3,183333	0,036145	0,032806	0,047544
	2	3500	3,874456	3,724631	0,039712	0,042154	0,04599
	2	4000	3,168621	3,36396	0,035071	0,039079	0,051308
L2	2	8000	1,911544	1,924526	0,084436	0,056686	0,076888
	2	13000	2,542896	2,534707	0,166597	0,202814	0,12751
	2	18000	2,124932	2,110889	0,186961	0,215129	0,129145
	2	23000	2,237994	1,949952	0,228131	0,287645	0,289931
	2	28000	1,952013	1,730564	0,230148	0,308019	0,285531
L3	2	33000	2,402191	2,186102	0,362633	0,462818	0,259251
	2	38000	1,58754	1,451468	0,297207	0,394157	0,36242
	2	43000	1,530781	1,567709	0,368821	0,486027	0,490784
	2	48000	2,093034	2,45275	0,561456	0,703775	0,711169
	2	53000	1,95948	1,9852	0,504974	0,625849	0,638028
	2	58000	2,243772	2,239115	0,675974	0,892505	0,875095
	2	100000	1,731284	1,99818	0,735864	0,851581	0,746933
	2	175000	2,261285	2,240408	1,083375	1,363663	1,553539
	2	250000	2,207516	2,135077	1,295233	1,650833	1,935149
	2	325000	1,916871	1,902564	1,31882	1,724247	1,749642
RAM	2	400000	1,534802	1,594194	1,369892	1,734846	1,643743
	2	475000	1,361007	1,619016	1,299292	1,631152	1,658817
	2	550000	2,148061	2,044756	1,106924	1,895393	1,234151
	2	625000	1,391836	1,452137	1,315637	1,389929	1,438449
	2	700000	1,382206	1,367695	1,293929	1,434509	1,469238
	2	775000	1,769168	1,764572	1,860003	2,05864	1,683302
HT	4	100000	1,577836	1,853299	0,578321	0,723271	0,805998
	4	175000	2,376492	2,299423	1,107256	1,624433	1,75382
	4	250000	1,952191	2,074723	0,950944	1,573093	1,595221
	4	325000	1,749904	1,669536	1,01724	1,827909	1,831662
	4	400000	1,566122	1,46625	1,233237	1,813515	1,795477
	4	475000	1,417551	1,471758	1,507163	1,876042	1,807037
	4	550000	1,862169	1,884381	2,00172	1,970697	2,481077
	4	625000	1,377995	1,445781	1,481438	1,570155	1,637799
	4	700000	1,965608	2,086574	2,279337	2,330487	2,291496
	4	775000	1,353648	1,428165	1,481366	1,55955	1,532064



	th	n	sse	paralelo	par-sse
L1	4	500	3,552542	0,00366	0,003816
	4	1000	3,757353	0,007821	0,008352
	4	1500	3,840142	0,012541	0,012011
	4	2000	3,871033	0,018678	0,018544
	4	2500	3,893733	0,022358	0,025674
	4	3000	3,918989	0,021399	0,022302
	4	3500	3,912618	0,031097	0,030231
	4	4000	3,894482	0,0323	0,029395
L2	4	8000	2,136004	0,060064	0,059526
	4	13000	1,981317	0,129572	0,138008
	4	18000	2,036807	0,138406	0,140221
	4	23000	2,14642	0,170985	0,180014
	4	28000	2,081444	0,171046	0,231085
	4	33000	2,161635	0,228537	0,23771
	4	38000	2,121378	0,299302	0,27128
	4	43000	2,176611	0,278612	0,277666
	4	48000	2,198296	0,367855	0,400587
	4	53000	2,179635	0,349724	0,349646
	4	58000	2,179401	0,436225	0,508782
	4	100000	2,184041	0,467928	0,465677
	4	175000	2,209455	0,537298	0,532585
	4	250000	2,000549	0,821162	0,735353
	4	325000	2,974956	0,849762	0,740722
	4	400000	2,191195	0,44084	0,735655
4	475000	2,170266	0,617689	0,461913	
4	550000	2,807621	0,890341	0,699461	
4	625000	1,786235	0,976024	0,50253	
4	700000	1,126599	0,454641	0,466298	
RAM	4	775000	1,677518	0,640132	0,77821
		1600000	1,02914	1,381808	1,429677



scopy

```

3 void scopy_seq(int n, float *x, float *y)
4 {
5     int i;
6     for (i = 0; i < n; ++i)
7         y[i] = x[i];
8 }

```

```

10 static void alig_scopy_sse(int n, float *x, float *y, int nprol, int nvecl)
11 {
12     int i;
13     → __m128 v_x, v_a;
14
15     //Prologo
16     for(i = 0; i < nprol ; i++)
17         y[i] = x[i];
18
19     //Cuerpo
20     for (i = nprol; i < nvecl; i+=4){
21     → v_x = mm load ps(&x[i]);
22     → _mm_store_ps(&y[i],v_x);
23     }
24
25     //Epilogo
26     for(i = nvecl; i < n ; i++)
27         y[i] = x[i];
28 }

```

```

6 static void alig_scopy_avx(int n, float *x, float *y, int nprol, int nvecl)
7 {
8     int i;
9     → __m256 v_x, v_y, v_a;
10
11     //Prologo
12     for(i = 0; i < nprol ; i++)
13         y[i] = x[i];
14
15     //Cuerpo
16     for (i = nprol; i < nvecl; i+=8){
17     → v_x = _mm256_load_ps(&x[i]);
18     → _mm256_store_ps(&y[i],v_x);
19     }
20
21     //Epilogo
22     for(i = nvecl; i < n ; i++)
23         y[i] = x[i];
24 }

```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	2.782548	2.569054	0.004287	0.005049	0.004195
4	1000	3.680454	3.769103	0.008824	0.008518	0.008630
4	1500	3.458409	3.267929	0.012908	0.013246	0.013674
4	2000	3.581439	3.631301	0.014762	0.017804	0.016779
4	2500	3.679113	3.794224	0.020737	0.018726	0.019774
4	3000	3.725955	3.743216	0.018284	0.023351	0.021359
4	3500	3.764521	3.178787	0.025645	0.023068	0.030061
4	4000	3.049088	2.929355	0.028023	0.028260	0.028690
4	8000	1.928038	1.943178	0.059991	0.075049	0.081476
4	13000	2.071354	2.053335	0.099145	0.108176	0.105700
4	18000	2.148229	2.158287	0.135206	0.146273	0.147017
4	23000	2.215422	1.936162	0.196664	0.189417	0.187455
4	28000	2.221615	2.182515	0.204854	0.220119	0.216328
4	33000	2.253448	2.219257	0.262776	0.257005	0.251099
4	38000	1.985321	1.853558	0.316889	0.316243	0.328125
4	43000	1.705495	1.744341	0.422200	0.396225	0.316220
4	48000	1.637848	1.316333	0.443130	0.378043	0.396150
4	53000	1.525535	1.612370	0.368824	0.457414	0.386152
4	58000	1.453100	1.527547	0.507855	0.395877	0.473819
4	100000	1.523344	1.501175	0.663959	0.854633	0.838751
4	175000	1.540294	1.545279	0.926394	1.154129	1.127571
4	250000	1.555452	1.573852	1.222877	1.518480	1.486962
4	325000	1.532363	1.455486	1.149285	1.531155	1.761399
4	400000	1.589577	1.617642	1.366744	1.818135	2.007509
4	475000	1.586020	1.574605	1.485247	2.226642	1.658453
4	550000	1.569091	1.581364	1.581790	2.339456	2.054733
4	625000	1.419214	1.456358	1.565135	1.792556	2.112182
4	700000	1.563955	1.617328	1.686761	2.304258	2.740108
4	775000	1.511478	1.532859	1.703029	2.117079	2.244655
4	1600000	1.260819	1.358395	1.664136	1.963800	2.407919
8	325000	1.513720	1.565863	1.318066	1.599668	1.519593
8	400000	1.487215	1.516737	1.361214	1.838419	1.789645
8	475000	1.557652	1.549360	1.662057	1.707527	1.794934
8	550000	1.544511	1.568698	1.561123	2.024971	1.907163
8	625000	1.508287	1.370709	1.560805	1.896369	1.868044
8	700000	1.493525	1.501153	1.793331	2.342410	2.223591
8	775000	1.505329	1.389168	2.099895	2.503371	2.142060
8	1600000	1.282840	1.132312	1.835608	2.247239	2.197293



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	2,759511	2,682959	0,003416	0,004178	0,004174
	2	1000	3,028791	3,8867	0,00788	0,009693	0,009698
	2	1500	3,627381	3,638209	0,011153	0,013105	0,018998
	2	2000	5,865638	3,443112	0,023026	0,022435	0,026359
	2	2500	4,881074	5,354688	0,044622	0,043153	0,0465
	2	3000	3,7279	3,615041	0,023505	0,028783	0,01749
	2	3500	3,515232	3,047023	0,030768	0,032477	0,027953
	2	4000	3,630816	3,628528	0,062561	0,045614	0,065482
L2	2	8000	1,90073	1,892238	0,0813	0,089985	0,083662
	2	13000	3,034552	3,094567	0,243489	0,271554	0,273971
	2	18000	2,956578	3,098526	0,191554	0,286057	0,280111
	2	23000	3,359668	3,183397	0,361453	0,361149	0,307285
	2	28000	2,221659	1,991591	0,208678	0,306313	0,3119
L3	2	33000	1,879365	1,583094	0,261059	0,355368	0,360049
	2	38000	1,583048	1,432454	0,32922	0,325541	0,382005
	2	43000	1,605934	1,714783	0,375055	0,376283	0,470417
	2	48000	1,483188	1,567112	0,390177	0,474078	0,481534
	2	53000	1,715287	2,012272	0,386725	0,543705	0,6469
	2	58000	1,52444	1,574623	0,489303	0,604691	0,609539
	2	100000	1,441842	1,396833	0,508968	0,558645	0,566669
	2	175000	1,974985	1,975414	0,820998	1,251682	1,174034
	2	250000	1,458951	1,775393	0,981743	1,206376	1,166167
RAM	2	325000	1,819119	1,718745	1,216335	1,497367	1,534603
	2	400000	1,362202	1,574487	1,345191	1,678174	1,626844
	2	475000	1,468481	1,579128	1,204453	1,470277	1,45485
	2	550000	1,513304	1,559257	1,288224	1,508633	1,577411
	2	625000	1,409165	1,453915	1,32732	1,666929	1,582101
	2	700000	1,795685	1,917596	1,777914	2,072554	1,941756
	2	775000	1,416947	1,415934	1,267269	1,346633	1,42265



	th	n	sse	paralelo	par-sse
L1	4	500	3,515294	0,004693	0,00491
	4	1000	3,749543	0,007793	0,007613
	4	1500	3,815231	0,017069	0,018904
	4	2000	3,875836	0,014904	0,016127
	4	2500	3,902703	0,019223	0,018624
	4	3000	3,91104	0,027904	0,027847
	4	3500	3,895127	0,03254	0,032348
	4	4000	3,901163	0,029947	0,029962
L2	4	8000	2,104712	0,05613	0,060412
	4	13000	2,01584	0,117768	0,112363
	4	18000	2,134774	0,133585	0,135568
	4	23000	2,154674	0,218256	0,237826
	4	28000	2,085288	0,299335	0,279027
	4	33000	2,338643	0,320019	0,327277
	4	38000	2,172425	0,245518	0,256225
	4	43000	2,113307	0,337079	0,328153
	4	48000	2,081546	0,251504	0,333881
	4	53000	2,173422	0,384409	0,426811
	4	58000	2,186963	0,359606	0,323627
	4	100000	2,188269	0,591477	0,730023
	4	175000	2,205375	0,657469	0,440221
	4	250000	2,360944	0,844656	0,889085
	4	325000	2,194365	0,533088	0,608499
	4	400000	3,052331	0,813084	0,90985
4	475000	1,450742	0,799934	0,626983	
4	550000	1,021098	0,90231	0,59192	
4	625000	1,954268	0,766045	0,574661	
4	700000	2,283705	0,755882	0,805577	
RAM	4	775000	1,736434	0,641595	0,744484
	4	1600000	1,032752	1,469478	1,677999



dcopy

```

3 void dcopy_seq(int n, double *x, double *y)
4 {
5     int i;
6     for (i = 0; i < n; ++i)
7         y[i] = x[i];
8 }

```

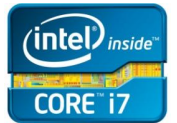
```

10 static void align_dcopy_sse(int n, double *x, double *y, int npr1, int nvec1)
11 {
12     int i;
13     __m128d v_x, v_a;
14
15     //Prologo
16     for(i = 0; i < npr1 ; i++)
17         y[i] = x[i];
18
19     //Cuerpo
20     for (i = npr1; i < nvec1; i+=2){
21         v_x = _mm_load_pd(&x[i]);
22         _mm_store_pd(&y[i],v_x);
23     }
24
25     //Epilogo
26     for(i = nvec1; i < n ; i++)
27         y[i] = x[i];
28 }
29
30 static void noalign_dcopy_sse(int n, double *x, double *y, int npr1, int nvec1)
31 {
32     int i;
33     __m128d v_x, v_a;
34
35     //Prologo
36     for(i = 0; i < npr1 ; i++)
37         y[i] = x[i];
38
39     //Cuerpo
40     for (i = npr1; i < nvec1; i+=2){
41         v_x = _mm_load_pd(&x[i]);
42         _mm_storeu_pd(&y[i],v_x);
43     }
44
45     //Epilogo
46     for(i = nvec1; i < n ; i++)
47         y[i] = x[i];
48 }
49 static void dcopy_sse(int n, double *x, double *y)
50 {
51
52     int dir1 = (int) (&x[0]);
53     int npr1 = 8 - ((dir1>>3)%8);
54     int nvec1 = (((int) floor((double)((n-npr1)/2.0))<<1)+ npr1);
55
56     int dir2 = (int) (&y[0]);
57     int npr2 = 8 - ((dir2>>3)%8);
58
59     if(npr1 == 8) npr1 = 0;
60     if(npr2 == 8) npr2 = 0;
61
62     if(npr1 == npr2)
63         align_dcopy_sse(n,x,y, npr1, nvec1);
64     else
65         noalign_dcopy_sse(n,x,y, npr1, nvec1);
66 }

```

```
6 static void align_dcopy_avx(int n, double *x, double *y, int npr1, int nvecl)
7 {
8     int i;
9     __m256d v_x, v_y, v_a;
10
11     //Prologo
12     for(i = 0; i < npr1 ; i++)
13         y[i] = x[i];
14
15     //Cuerpo
16     for (i = npr1; i < nvecl; i+=4){
17         v_x = _mm256_load_pd(&x[i]);
18         _mm256_store_pd(&y[i],v_x);
19     }
20
21     //Epilogo
22     for(i = nvecl; i < n ; i++)
23         y[i] = x[i];
24 }
25
26 static void noalign_dcopy_avx(int n, double *x, double *y, int npr1, int nvecl)
27 {
28     int i;
29     __m256d v_x, v_y, v_a;
30
31     //Prologo
32     for(i = 0; i < npr1 ; i++)
33         y[i] = x[i];
34
35     //Cuerpo
36     for (i = npr1; i < nvecl; i+=4){
37         v_x = _mm256_load_pd(&x[i]);
38         _mm256_storeu_pd(&y[i],v_x);
39     }
40
41     //Epilogo
42     for(i = nvecl; i < n ; i++)
43         y[i] = x[i];
44 }
45 static void dcopy_avx(int n, double *x, double *y)
46 {
47
48     int dir1 = (int) (&x[0]);
49     int npr1 = 8 - ((dir1>>3)%8);
50     int nvecl = (((int) floor((double)((n-npr1)/4.0)))<<2) + npr1;
51
52     int dir2 = (int) (&y[0]);
53     int npr2 = 8 - ((dir2>>3)%8);
54
55     if(npr1 == 8) npr1 = 0;
56     if(npr2 == 8) npr2 = 0;
57
58     if(npr1 == npr2)
59         align_dcopy_avx(n,x,y, npr1, nvecl);
60     else
61         noalign_dcopy_avx(n,x,y, npr1, nvecl);
62 }
```


TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	1.936597	1.984375	0.004964	0.005621	0.004373
4	1000	1.810353	1.845969	0.009676	0.009923	0.008084
4	1500	1.841346	1.910542	0.013537	0.013329	0.013330
4	2000	1.607605	1.538681	0.017456	0.017716	0.018248
4	2500	1.221178	1.195986	0.026853	0.020111	0.029278
4	3000	1.143797	1.160394	0.030605	0.028494	0.030315
4	3500	1.114764	1.116470	0.035743	0.032082	0.039493
4	4000	1.145927	1.147613	0.050361	0.037472	0.038728
4	8000	1.093135	1.212534	0.076910	0.071198	0.072064
4	13000	1.242104	1.247849	0.105613	0.116338	0.129984
4	18000	1.186684	1.194310	0.211096	0.219884	0.174964
4	23000	1.154166	1.146907	0.255756	0.255546	0.259877
4	28000	1.094462	1.027315	0.335130	0.283670	0.362523
4	33000	1.080951	1.019003	0.360949	0.347764	0.327393
4	38000	1.041691	1.036313	0.395022	0.402124	0.364380
4	43000	1.085357	1.142311	0.474496	0.447626	0.495391
4	48000	1.076835	1.042364	0.469867	0.613349	0.556224
4	53000	1.084767	1.082809	0.542682	0.553429	0.664817
4	58000	1.128884	1.138391	0.635857	0.638978	0.690246
4	100000	1.064998	1.077885	0.873460	0.891095	0.883754
4	175000	1.073255	1.080767	1.080189	1.124272	1.099620
4	250000	1.050083	1.040330	1.188114	1.200799	0.987307
4	325000	1.069244	1.091074	1.440675	1.423125	1.655184
4	400000	1.013572	1.058726	1.464848	1.505119	1.507761
4	475000	1.154572	0.991334	1.630795	1.723203	1.793848
4	550000	1.012223	0.967739	1.414647	1.445245	1.469199
4	625000	1.059496	0.996636	1.503949	1.574771	1.516574
4	700000	1.027836	0.992946	1.453459	1.562184	1.595559
4	775000	1.042235	1.086717	1.595807	1.723584	1.650584
4	1600000	1.037392	1.042642	1.405261	1.673527	1.880507
8	400000	1.020998	0.835904	1.193186	1.516546	1.614462
8	475000	1.125989	1.118240	1.917817	2.058287	2.051958
8	550000	1.038512	0.995342	1.473220	1.594398	1.833537
8	625000	1.021968	0.992925	1.857015	1.939610	1.977166
8	700000	1.116424	1.103595	1.846663	1.927872	1.907542
8	775000	1.078407	1.026919	1.650971	1.819266	2.036654
8	1600000	1.093299	1.113734	2.090452	2.085236	2.154170



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	1,976898	2,021941	0,00294	0,005425	0,007328
	2	1000	1,732199	1,859615	0,009849	0,011842	0,011893
	2	1500	1,846792	1,921621	0,017267	0,01775	0,017811
	2	2000	1,650703	1,539631	0,023509	0,024399	0,024514
L2	2	2500	1,14836	1,160011	0,030883	0,03736	0,027089
	2	3000	1,152819	1,171178	0,037975	0,04425	0,044726
	2	3500	1,115838	1,11769	0,052245	0,054084	0,047314
	2	4000	1,142204	1,147387	0,050418	0,061689	0,054622
	2	8000	1,095202	1,210071	0,096901	0,11133	0,112297
	2	13000	1,232046	1,244349	0,135228	0,159912	0,164095
L3	2	18000	1,118323	1,174794	0,171444	0,203154	0,216503
	2	23000	1,149881	1,143615	0,293043	0,327171	0,330178
	2	28000	1,082235	1,066076	0,323656	0,356698	0,355666
	2	33000	1,06052	1,058001	0,343678	0,376552	0,375818
	2	38000	1,060427	1,033005	0,403027	0,443352	0,431126
	2	43000	1,123658	1,129761	0,465793	0,514714	0,517741
	2	48000	1,076178	1,071734	0,399825	0,437859	0,399236
	2	53000	1,008347	1,053313	0,486452	0,522122	0,524003
	2	58000	0,95528	1,057979	0,436891	0,433026	0,442638
	2	100000	1,003257	1,057849	0,571428	0,612965	0,622486
RAM	2	175000	1,068152	1,026713	0,8268	0,883954	0,935499
	2	250000	1,024836	1,071511	1,093661	1,070312	1,083801
	2	325000	1,05405	1,109329	1,154659	1,121735	1,128786
	2	400000	0,95753	1,050815	1,059062	1,091958	1,184231
	2	475000	1,034132	1,039424	1,255731	1,27658	1,194203
	2	550000	1,029317	1,019192	1,179379	1,200411	1,204843
	2	625000	1,045599	1,028086	1,181467	1,240683	1,253159
	2	700000	1,065995	1,039964	1,275084	1,236041	1,267786
	2	775000	1,150008	1,157529	1,474691	1,54141	1,418312



	th	n	sse	paralelo	par-sse
L1	4	500	2,181319	0,004763	0,004978
	4	1000	2,86862	0,010161	0,010465
	4	1500	1,979961	0,012273	0,014216
	4	2000	1,984386	0,018577	0,019828
L2	4	2500	1,864843	0,036676	0,036323
	4	3000	1,844834	0,035295	0,035949
	4	3500	1,829999	0,059988	0,059301
	4	4000	1,813918	0,050283	0,044968
	4	8000	1,903267	0,113692	0,10546
	4	13000	1,921761	0,187191	0,20365
	4	18000	1,926988	0,220469	0,23735
	4	23000	1,932816	0,355301	0,376065
	4	28000	2,570382	0,490545	0,474156
	4	33000	1,924755	0,360375	0,404419
	4	38000	1,925163	0,397974	0,416342
	4	43000	1,920826	0,363733	0,581296
	4	48000	1,934022	0,567024	0,568858
	4	53000	1,938457	0,54173	0,541301
	4	58000	1,894564	0,348682	0,370324
	4	100000	2,007215	0,617757	0,472796
4	175000	1,272942	0,440275	0,555841	
4	250000	1,335316	0,828185	0,77736	
4	325000	2,111957	0,547129	0,624309	
RAM	4	400000	0,654623	0,681223	0,587887
	4	475000	1,05063	1,51922	1,586086
	4	550000	0,99765	1,407229	1,220813
	4	625000	0,964298	1,346522	1,764511
	4	700000	1,003916	1,585859	1,807902
	4	775000	0,98377	1,610126	1,669583



Scal iscal.

```

1  // #include "var.h"
2
3  static void iscal_seq(int n, int a, int *x)
4  {
5      int i;
6      for (i = 0; i < n; ++i)
7          x[i] = a*x[i];
8

```

```

11 static void alig_iscal_sse(int n, int a, int *x, int nprol, int nvecl)
12 {
13     int i;
14     __m128i v_x, v_a;
15
16     //Prologo
17     for(i = 0; i < nprol ; i++)
18         x[i] = a*x[i];
19
20     //Cuerpo
21     v_a = _mm_set1_epi32(a);
22     for (i = nprol; i < nvecl; i+=4){
23         v_x = _mm_load_si128((__m128i *)&x[i]);
24         v_x = _mm_mullo_epi32(v_a, v_x);
25         _mm_store_si128((__m128i *)&x[i], v_x);
26     }
27
28     //Epilogo
29     for(i = nvecl; i < n ; i++)
30         x[i] = a*x[i];
31 }

```

```

34 static void iscal_sse(int n, int a, int *x)
35 {
36
37     {
38         int dir1 = (int) (&x[0]);
39         int nprol1 = 16 - ((dir1 >> 2) % 16);
40         int nvecl = (((int) floor((double) ((n - nprol1) / 4.0))) << 2) + nprol1;
41
42         if(nprol1 == 16) nprol1 = 0;
43
44         alig_iscal_sse(n, a, x, nprol1, nvecl);
45     }
46 }

```

```

10 static void iscal_omp(int n, int a, int *x)
11 {
12     int i;
13     #pragma omp parallel for
14     for (i = 0; i < n; ++i)
15         x[i] = a*x[i];
16 }
17 }
18
19 static void iscal_omp_sse(int n, int a, int *x)
20 {
21     int i,trozo,ini,fin;
22
23     trozo=ceil((float)n/(float)NTHR);
24
25     #pragma omp parallel private(i,ini,fin) firstprivate (a,n,trozo)
26     {
27         i = omp_get_thread_num();
28         ini=i*trozo;
29         fin=min(trozo*(i+1),n);
30         iscal_sse(fin-ini,a,&x[ini]);
31     }
32 }

```

1 #include <omp.h>

```

6 static void alig_iscal_avx(int n, int a, int *x, int nprol, int nvecl)
7 {
8     int i;
9     __m256i v_x, v_a;
10
11     //Prologo
12     for(i = 0; i < nprol ; i++)
13         x[i] = a*x[i];
14
15     //Cuerpo
16     v_a = mm256_set1_epi32(a);
17     for (i = nprol; i < nvecl; i+=8){
18         v_x = mm256_load_si256((__m256i *)&x[i]);
19         → v_x = _mm256_mullo_epi32(v_a,v_x);
20         _mm256_store_si256((__m256i *)&x[i],v_x);
21     }
22
23     //Epilogo
24     for(i = nvecl; i < n ; i++)
25         x[i] = a*x[i];
26 }

```

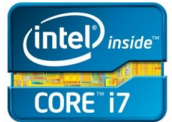
1 #include <immintrin.h> //AVX

```

28 static void iscal_avx(int n, int a, int *x)
29 {
30
31     int dir1 = (int) (&x[0]);
32     int nprol1 = 16 - ((dir1>>2)%16);
33     int nvecl = (((int) floor((double)((n-nprol1)/8.0))<<3)+ nprol1;
34
35     if(nprol1 == 16) nprol1 = 0;
36
37     alig_iscal_avx(n,a,x, nprol1, nvecl);
38 }

```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	2,450475	4,630769	0,004627	0,00521	0,004659
4	1000	2,567892	4,794118	0,010412	0,011025	0,009591
4	1500	4,377676	3,087347	0,026494	0,019237	0,024532
4	2000	2,583017	4,849822	0,015882	0,017358	0,018984
4	2500	2,761473	5,157434	0,028942	0,034377	0,027788
4	3000	4,681723	9,092799	0,044794	0,048235	0,046697
4	3500	2,626229	5,189845	0,029725	0,03014	0,031826
4	4000	3,150265	6,301273	0,063545	0,06479	0,071282
4	8000	2,723679	5,358612	0,080546	0,083069	0,0785
4	13000	2,534609	3,524688	0,115745	0,096797	0,14458
4	18000	2,576256	3,498235	0,161333	0,158524	0,166306
4	23000	3,731185	5,151412	0,333616	0,291005	0,30242
4	28000	2,547399	3,51366	0,24164	0,215888	0,301322
4	33000	2,554763	3,52641	0,262502	0,289186	0,303829
4	38000	2,533272	3,384197	0,314402	0,367552	0,342944
4	43000	2,330823	3,234356	0,31512	0,314295	0,337215
4	48000	2,565668	3,492594	0,458828	0,411507	0,410979
4	53000	2,44689	3,379527	0,412035	0,407401	0,420204
4	58000	2,564256	3,523811	0,465259	0,465014	0,490345
4	100000	2,320869	2,672785	0,730621	0,725424	0,914142
4	175000	2,189118	1,8683	0,899299	1,216816	1,305967
4	250000	2,093406	2,452184	1,005671	1,387922	1,713751
4	325000	2,152887	2,48105	1,220579	2,023724	2,199682
4	400000	2,236162	2,587881	1,27512	2,361952	2,352546
4	475000	2,080252	2,434529	1,260497	2,250971	2,711744
4	550000	2,064453	2,315051	1,543429	1,744811	2,15351
4	625000	2,107032	2,373171	1,422059	2,622981	2,876465
4	700000	2,185817	2,506727	1,48679	3,304083	3,750451
4	775000	1,534847	1,802454	1,65259	2,65213	3,050269
4	1600000	1,852318	2,251696	1,820483	3,118804	3,470367



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	2,505429	4,444834	0,006976	0,007821	0,00783
	2	1000	2,568827	4,767396	0,007606	0,009688	0,009626
	2	1500	2,619805	4,948123	0,01704	0,013216	0,014134
	2	2000	2,616106	5,094522	0,01463	0,018474	0,027098
	2	2500	2,333527	3,325806	0,023879	0,021587	0,024137
	2	3000	3,607563	6,662125	0,081812	0,075608	0,09099
	2	3500	3,753832	5,515766	0,045622	0,042707	0,046593
	2	4000	1,91253	5,178981	0,050595	0,039556	0,050913
	2	8000	2,67978	5,006765	0,089942	0,082683	0,115992
L2	2	13000	2,441001	4,276368	0,18356	0,266542	0,161144
	2	18000	3,133613	4,714325	0,239469	0,187394	0,234286
	2	23000	1,611432	3,042088	0,224652	0,322128	0,333794
	2	28000	2,974758	5,253649	0,405961	0,481392	0,664582
	2	33000	2,89681	5,324537	0,560225	0,78097	0,828861
	2	38000	3,057773	3,344093	0,430494	0,567679	0,602972
	2	43000	3,006734	4,459216	0,514357	0,689047	0,733173
	2	48000	1,745467	2,763688	0,422229	0,578143	0,61917
	2	53000	1,93936	2,510523	0,415839	0,464694	0,489861
L3	2	58000	2,412715	3,860321	0,634285	0,806088	0,89981
	2	100000	2,915038	3,332258	0,623832	0,746441	1,002534
	2	175000	2,840057	4,107424	1,327943	2,004768	2,068097
	2	250000	2,617268	2,82799	1,031244	1,427822	1,544863
	2	325000	3,634559	4,074858	1,643208	2,332277	2,59692
	2	400000	2,789091	3,032267	1,398916	2,177236	2,345482
	2	475000	2,773548	3,444782	1,65739	2,737854	3,120337
	2	550000	3,742708	4,798623	2,389868	4,69417	4,998923
	2	625000	1,803322	2,402735	1,487585	1,884737	2,303104
2	700000	2,078207	2,418471	1,279319	2,366782	2,576625	
2	775000	2,080001	2,475509	1,121982	1,897843	2,253634	



	th	n	sse	paralelo	par-sse
L1	4	500	2,711667	0,004886	0,00488
	4	1000	2,823529	0,009407	0,010314
	4	1500	2,756596	0,011051	0,012322
	4	2000	2,798976	0,015508	0,015907
	4	2500	2,707468	0,023991	0,025294
	4	3000	2,894783	0,024355	0,036413
	4	3500	2,895334	0,035226	0,032759
	4	4000	2,667759	0,029973	0,030027
	4	8000	2,688948	0,070881	0,078588
L2	4	13000	2,284931	0,144476	0,140331
	4	18000	2,316779	0,150533	0,133975
	4	23000	2,23743	0,180603	0,17347
	4	28000	2,318115	0,217501	0,210054
	4	33000	2,305346	0,286969	0,315711
	4	38000	2,307998	0,279379	0,288157
	4	43000	2,352703	0,300844	0,359245
	4	48000	2,310196	0,277935	0,32499
	4	53000	2,328421	0,391668	0,450094
	4	58000	2,686762	0,518282	0,610229
	4	100000	2,316226	0,444961	0,569662
	4	175000	2,859558	0,587642	0,901864
	4	250000	2,739374	0,759154	1,003252
	4	325000	1,602521	0,701784	0,954171
	4	400000	2,339972	0,704966	0,998622
	4	475000	2,310021	0,724606	0,994005
	4	550000	2,415017	0,69908	1,031605
4	625000	2,30059	0,673164	0,942645	
4	700000	2,300069	0,768593	0,813167	
4	775000	2,108794	0,736669	0,928442	
RAM	4	1600000	1,501847	1,141754	1,266951



sscal.

```

10 static void align_sscal_sse(int n, float a, float *x, int nprol, int nvecl)
11 {
12     int i;
13     __m128 v_x, v_a;
14
15     //Prologo
16     for(i = 0; i < nprol ; i++)
17         x[i] = a*x[i];
18
19     //Cuerpo
20     v_a = _mm_set1_ps(a);
21     for (i = nprol; i < nvecl; i+=4){
22         v_x = _mm_load_ps(&x[i]);
23         v_x = _mm_mul_ps(v_a,v_x);
24         _mm_store_ps(&x[i],v_x);
25     }
26
27     //Epilogo
28     for(i = nvecl; i < n ; i++)
29         x[i] = a*x[i];
30 }

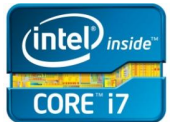
```

```

6 static void align_sscal_avx(int n, float a, float *x, int nprol, int nvecl)
7 {
8     int i;
9     __m256 v_x, v_a;
10
11     //Prologo
12     for(i = 0; i < nprol ; i++)
13         x[i] = a*x[i];
14
15     //Cuerpo
16     v_a = mm256_set1_ps(a);
17     for (i = nprol; i < nvecl; i+=8){
18         v_x = mm256_load_ps(&x[i]);
19         v_x = _mm256_mul_ps(v_a,v_x);
20         _mm256_store_ps(&x[i],v_x);
21     }
22
23     //Epilogo
24     for(i = nvecl; i < n ; i++)
25         x[i] = a*x[i];
26 }

```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	4,361146	8,16317	0,006636	0,007898	0,006943
4	1000	3,516551	5,841432	0,008541	0,00905	0,008587
4	1500	3,247596	6,408918	0,012697	0,011308	0,014539
4	2000	3,505439	6,445	0,01885	0,019798	0,018629
4	2500	3,327203	6,551402	0,023617	0,022496	0,023101
4	3000	3,53442	6,53445	0,031002	0,032694	0,032178
4	3500	3,279545	6,748158	0,035903	0,032453	0,032608
4	4000	3,637175	6,606708	0,033624	0,036582	0,030067
4	8000	3,238249	6,38486	0,06775	0,062564	0,067109
4	13000	3,409028	3,527637	0,160682	0,081073	0,131711
4	18000	3,074762	3,140739	0,146906	0,148373	0,144889
4	23000	3,417806	3,495215	0,208003	0,204648	0,210781
4	28000	3,425509	3,510491	0,238938	0,229581	0,249771
4	33000	3,343009	3,394014	0,279566	0,27528	0,28402
4	38000	2,946564	3,37266	0,315616	0,318759	0,329361
4	43000	3,47736	3,599125	0,379696	0,369455	0,379056
4	48000	3,32961	3,471755	0,322043	0,39221	0,399837
4	53000	3,226222	3,500502	0,407345	0,396348	0,487422
4	58000	8,125531	7,182925	1,134439	1,11969	1,11147
4	100000	2,332454	2,511044	0,650735	0,718305	0,767161
4	175000	2,677294	2,905328	0,975819	1,447864	1,424506
4	250000	2,506884	2,761831	1,02169	1,719871	1,788194
4	325000	2,299768	2,686119	1,162646	2,106734	2,137559
4	400000	3,059054	3,559754	1,47626	2,204163	2,165431
4	475000	2,838837	3,106921	1,45128	2,865976	0,876459
4	550000	2,287528	2,552707	1,365897	3,237935	2,693821
4	625000	2,341332	2,329308	1,414213	2,739	2,876423
4	700000	2,306477	2,441254	1,450383	2,200428	3,034939
4	775000	2,181987	2,144166	1,503445	3,119424	2,950854
4	1600000	2,123277	2,211195	1,674949	3,117274	3,693655




	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	3,686937	6,304236	0,012417	0,015093	0,014965
	2	1000	3,349527	5,633406	0,022745	0,023664	0,023682
	2	1500	6,15519	11,07024	0,030612	0,024186	0,022467
	2	2000	3,522423	6,829901	0,024184	0,023688	0,014026
	2	2500	3,517544	6,505214	0,032205	0,02825	0,019278
	2	3000	3,615551	3,658017	0,023907	0,021552	0,027779
	2	3500	3,535739	6,786585	0,040886	0,047213	0,04744
	2	4000	3,637447	6,999223	0,047529	0,033253	0,036629
	2	8000	3,634567	7,078973	0,082002	0,101135	0,079564
L2	2	13000	3,016587	3,10136	0,127139	0,125849	0,168955
	2	18000	3,558963	6,43917	0,297173	0,402246	0,31634
	2	23000	3,478154	5,538863	0,254331	0,556305	0,430972
	2	28000	3,203341	4,968202	0,378784	0,470889	0,370729
	2	33000	3,384868	5,771006	0,595564	0,663442	0,562296
	2	38000	3,585926	6,084005	0,563568	0,550837	0,693642
	2	43000	3,428265	5,714246	0,631158	0,824343	0,836008
	2	48000	3,751631	5,766245	0,628015	0,922429	0,951593
	2	53000	3,353917	4,760774	0,656493	0,687952	0,9264
2	58000	3,487665	5,13512	0,862683	1,212835	1,135452	
L3	2	100000	3,304803	4,343615	0,097478	0,440652	1,870851
	2	175000	3,199377	4,156104	1,33285	1,836022	1,8795
	2	250000	2,814232	3,210891	1,06556	1,59211	1,728247
	2	325000	2,687795	3,159985	1,378212	2,118942	2,152329
	2	400000	2,79528	2,957897	1,3716	2,126901	2,11943
	2	475000	2,428865	2,777099	1,059344	1,971831	2,053455
	2	550000	2,452348	2,662047	1,248345	2,093357	2,312795
	2	625000	3,088206	4,313898	2,017398	3,851822	4,246836
	2	700000	2,555384	2,740418	1,508204	2,623398	2,735724
2	775000	2,364445	2,550136	1,505635	2,985815	2,855299	



	th	n	sse	paralelo	par-sse
L1	4	500	3,462295	0,003755	0,004113
	4	1000	3,667506	0,0097	0,009867
	4	1500	3,949094	0,014791	0,01565
	4	2000	3,752611	0,019227	0,019519
	4	2500	3,805723	0,019161	0,018896
	4	3000	3,835339	0,024087	0,025652
	4	3500	3,840953	0,031995	0,033405
	4	4000	3,833389	0,036234	0,038263
	4	8000	3,991924	0,078666	0,078778
L2	4	13000	2,863146	0,090999	0,093719
	4	18000	2,938515	0,174844	0,160604
	4	23000	2,955578	0,224392	0,250892
	4	28000	2,964048	0,246937	0,263694
	4	33000	2,899415	0,28795	0,296806
	4	38000	2,937219	0,30522	0,367557
	4	43000	2,898499	0,302117	0,330373
	4	48000	3,959682	0,437668	0,499356
	4	53000	2,984778	0,340865	0,402351
	4	58000	3,006833	0,365346	0,439016
	4	100000	2,841785	0,557897	0,744235
	4	175000	3,216861	0,638151	0,803514
	4	250000	2,929365	0,502562	0,841858
	4	325000	3,821372	0,788566	1,288782
	4	400000	2,835165	0,686218	1,008598
	4	475000	2,416189	0,960573	1,275849
	4	550000	2,686743	0,747341	0,963391
4	625000	1,777053	0,797373	1,012236	
4	700000	2,397174	0,773987	0,952328	
4	775000	2,842341	0,739055	1,066898	
RAM	4	1600000	1,252503	1,191709	1,291794



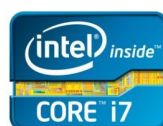
dscal.



```
9 static void align_dscal_sse(int n, double a, double *x, int nprol, int nvecl)
10 {
11     int i;
12     __m128d v_x, v_a;
13
14     //Prologo
15     for(i = 0; i < nprol ; i++)
16         x[i] = a*x[i];
17
18     //Cuerpo
19     v_a = _mm_set1_pd(a);
20     for (i = nprol; i < nvecl; i+=2){
21         v_x = mm_load_pd(&x[i]);
22         v_x = _mm_mul_pd(v_a,v_x);
23         _mm_store_pd(&x[i],v_x);
24     }
25
26     //Epilogo
27     for(i = nvecl; i < n ; i++)
28         x[i] = a*x[i];
29 }
```

```
6 static void align_dscal_avx(int n, double a, double *x, int nprol, int nvecl)
7 {
8     int i;
9     __m256d v_x, v_a;
10
11     //Prologo
12     for(i = 0; i < nprol ; i++)
13         x[i] = a*x[i];
14
15     //Cuerpo
16     v_a = _mm256_set1_pd(a);
17     for (i = nprol; i < nvecl; i+=4){
18         v_x = mm256_load_pd(&x[i]);
19         v_x = _mm256_mul_pd(v_a,v_x);
20         _mm256_store_pd(&x[i],v_x);
21     }
22
23     //Epilogo
24     for(i = nvecl; i < n ; i++)
25         x[i] = a*x[i];
26 }
```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	1,88516	3,16	0,002754	0,006195	0,004545
4	1000	1,769908	3,564671	0,008885	0,008885	0,010539
4	1500	1,864491	3,443105	0,013739	0,012906	0,013227
4	2000	1,777673	3,654946	0,019668	0,01757	0,017684
4	2500	1,873658	3,487402	0,022353	0,021333	0,021557
4	3000	1,776814	3,759244	0,032166	0,033343	0,021552
4	3500	1,941513	3,583228	0,032258	0,028283	0,031475
4	4000	1,601475	3,816221	0,038246	0,025787	0,036829
4	8000	1,839472	1,889917	0,076922	0,07767	0,07438
4	13000	1,832009	1,881184	0,121691	0,123885	0,135748
4	18000	1,852607	1,885201	0,170167	0,188787	0,17449
4	23000	1,577793	2,413471	0,206623	0,313859	0,348598
4	28000	1,87383	1,928987	0,243797	0,270596	0,24855
4	33000	1,657968	1,81078	0,296205	0,318074	0,30542
4	38000	1,646854	1,723155	0,347171	0,429462	0,433551
4	43000	1,431202	1,467515	0,222168	0,381342	0,37762
4	48000	1,387075	1,470433	0,411749	0,477916	0,491882
4	53000	1,433952	1,614994	0,441576	0,478365	0,530714
4	58000	1,34654	1,483028	0,532969	0,501156	0,536449
4	100000	1,441947	1,532887	0,725089	0,823273	0,850994
4	175000	1,460583	1,566856	0,916678	1,188954	1,284086
4	250000	1,393129	1,46305	1,066258	1,405201	1,41721
4	325000	1,429886	1,561967	1,26772	1,058732	1,748952
4	400000	1,472181	1,486765	1,691041	2,271918	2,23647
4	475000	1,36532	1,379423	1,433973	1,828617	1,969352
4	550000	1,378518	1,452616	1,301353	1,971194	2,020636
4	625000	1,349264	1,436578	1,397408	1,984976	2,100892
4	700000	1,207178	1,404447	1,312235	1,987774	2,050677
4	775000	1,358285	1,22379	1,568924	2,064922	2,229287
4	1600000	1,244455	1,207285	1,616915	1,970984	1,986037



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	1,945698	3,479193	0,006817	0,007211	0,007102
	2	1000	1,762966	2,226476	0,009536	0,014935	0,015007
	2	1500	1,959297	3,857278	0,018845	0,020022	0,024956
	2	2000	1,713226	3,463424	0,018959	0,02272	0,028404
	2	2500	1,831122	4,224323	0,06419	0,07531	0,068252
	2	3000	1,658334	3,463455	0,038283	0,042244	0,044322
	2	3500	1,822866	3,704887	0,053824	0,087241	0,088042
	2	4000	1,829401	3,100409	0,081299	0,08355	0,084197
L2	2	8000	1,193691	2,596755	0,161621	0,187858	0,191308
	2	13000	1,198137	2,010339	0,16909	0,221603	0,230431
	2	18000	1,796389	2,651284	0,266846	0,31366	0,330336
	2	23000	1,222284	1,549504	0,254403	0,29233	0,240096
	2	28000	1,214178	1,491392	0,269432	0,336275	0,350022
L3	2	33000	1,691546	1,813387	0,3169	0,386294	0,400697
	2	38000	1,232051	2,260463	0,497833	0,581354	0,595193
	2	43000	1,441457	1,756992	0,444883	0,545159	0,564373
	2	48000	1,214317	1,708878	0,377821	0,580628	0,603773
	2	53000	1,07416	1,579333	0,392837	0,483511	0,536706
	2	58000	1,295868	1,585705	0,530206	0,620207	0,585919
	2	100000	1,916151	1,715222	0,824745	1,038821	1,064314
	2	175000	1,66456	2,156129	1,271959	2,308506	2,557967
	2	250000	1,417671	1,409707	1,024743	1,462505	1,573025
	2	325000	1,117507	1,361344	1,161432	1,446602	1,803096
RAM	2	400000	1,272041	1,381496	0,939748	1,345473	1,620843
	2	475000	1,442185	1,616937	1,698357	1,341438	1,365913
	2	550000	1,382567	1,81956	1,708359	1,967431	1,576539
	2	625000	1,455165	1,547408	1,585743	1,386324	1,890598
	2	700000	1,064612	1,416282	1,409683	1,580497	1,60418
	2	775000	1,321423	1,338138	1,416434	1,587138	1,581681



	th	n	sse	paralelo	par-sse
L1	4	500	1,971759	0,004953	0,005143
	4	1000	1,939647	0,00868	0,008637
	4	1500	1,996122	0,011745	0,011493
	4	2000	1,957404	0,018515	0,018943
	4	2500	1,994189	0,024264	0,023739
	4	3000	1,979789	0,028824	0,02756
	4	3500	1,985793	0,033538	0,03346
	4	4000	1,955363	0,037712	0,038113
L2	4	8000	1,559181	0,08651	0,07592
	4	13000	1,928	0,130516	0,12707
	4	18000	1,5604	0,143992	0,158948
	4	23000	1,568425	0,226989	0,242982
	4	28000	1,40788	0,232475	0,252069
	4	33000	1,564396	0,239632	0,23247
	4	38000	1,543746	0,262729	0,264462
	4	43000	1,56201	0,2648	0,275013
	4	48000	1,57702	0,30161	0,297464
	4	53000	1,544348	0,28772	0,30468
	4	58000	1,558814	0,324894	0,323361
	4	100000	1,588376	0,374824	0,442503
	4	175000	1,541481	0,46725	0,515464
	4	250000	1,550348	0,497147	0,487341
	4	325000	1,517486	0,436334	0,565096
	4	400000	1,901837	0,580161	0,699527
4	475000	1,505023	0,569103	0,584754	
4	550000	1,733443	0,65992	0,622239	
4	625000	1,569652	0,437742	0,616967	
4	700000	1,12061	0,740426	0,798186	
RAM	4	775000	1,513003	1,119004	0,951573
	4	1600000	1,25621	1,773974	1,897858



dot

idot

```

4  int idot_seq(int n, int *x, int *y)
5  {
6      int i;
7      int dot = 0;
8      for (i = 0; i < n; ++i){
9          dot = dot + x[i]*y[i];
10     }
11
12     return dot;
13 }

```

```

11 static int alig_idot_sse(int n, int *x, int *y, int nprol, int nvecl)
12 {
13     int i,j;
14     int dot = 0;
15     int attribute__((aligned(64))) vv[4];
16     __m128i v_x, v_y, v_w;
17
18     //Prologo
19     for(i = 0; i < nprol ; i++)
20         dot = dot + x[i]*y[i];
21
22     //Cuerpo
23     v_w = _mm_setzero_si128();
24     for (i = nprol; i < nvecl; i+=4){
25         v_x = _mm_load_si128((__m128i *)&x[i]);
26         v_y = _mm_load_si128((__m128i *)&y[i]);
27         v_w = _mm_mullo_epi32(v_y,v_x);
28         v_w = _mm_add_epi32(v_y,v_w);
29     }
30     v_w = _mm_hadd_epi32(v_w,v_w);
31     v_w = _mm_hadd_epi32(v_w,v_w);
32     _mm_store_si128((__m128i *)&vv[0], v_w);
33     dot = dot + vv[0];
34
35     //Epilogo
36     for(i = nvecl; i < n ; i++)
37         dot = dot + x[i]*y[i];
38     return dot;
39 }

```

```

11 static int noalig_idot_sse(int n, int *x, int *y, int npr1, int nvec1)
12 {
13     int i,j;
14     int dot = 0;
15     int __attribute__((aligned(64))) vv[4];
16     __m128i v_x, v_y, v_w;
17
18     //Prologo
19     for(i = 0; i < npr1 ; i++)
20         dot = dot + x[i]*y[i];
21
22     //Cuerpo
23     v_w = mm setzero si128();
24     for (i = npr1; i < nvec1; i+=4){
25         v_x = _mm_loadu_si128((__m128i *)&x[i]);
26         v_y = mm load si128(( m128i *)&y[i]);
27         v_y = _mm_mullo_epi32(v_y,v_x);
28         v_w = _mm_add_epi32(v_y,v_w);
29     }
30     v_w = mm hadd_epi32(v_w,v_w);
31     v_w = _mm_hadd_epi32(v_w,v_w);
32     mm store si128((__m128i *)&vv[0], v_w);
33     dot = dot + vv[0];
34
35     //Epilogo
36     for(i = nvec1; i < n ; i++)
37         dot = dot + x[i]*y[i];
38     return dot;
39 }
40
41 static int idot_sse(int n, int *x, int *y)
42 {
43     int dot = 0;
44
45     int dir1 = (int) (&y[0]);
46     int npr1 = 16 - ((dir1>>2)%16);
47     int nvec1 = (((int) floor((double)((n-npr1)/4.0))<<2)+ npr1);
48
49     int dir2 = (int) (&x[0]);
50     int npr2 = 16 - ((dir2>>2)%16);
51
52     if(npr1 == 16) npr1 = 0;
53     if(npr2 == 16) npr2 = 0;
54
55     if(npr1 == npr2) dot = alig_idot_sse(n,x,y, npr1, nvec1);
56     else dot = noalig_idot_sse(n,x,y, npr1, nvec1);
57
58     return dot;
59 }

```

```

6 static int alig_idot_avx(int n, int *x, int *y, int npr1, int nvec1)
7 {
8     int i,j;
9     int dot = 0;
10    int attribute__((aligned(64))) vv[8];
11    __m256i v_x, v_y, v_w;
12
13    //Prologo
14    for(i = 0; i < npr1 ; i++)
15        dot = dot + x[i]*y[i];
16
17    //Cuerpo
18    v_w = mm256_setzero_si256();
19    for (i = npr1; i < nvec1; i+=8){
20        v_x = _mm256_load_si256((__m256i *)&x[i]);
21        v_y = mm256_load_si256((__m256i *)&y[i]);
22        → v_y = _mm256_mullo_epi32(v_y,v_x);
23        → v_w = _mm256_add_epi32(v_y,v_w);
24    }
25    → v_w = _mm256_hadd_epi32(v_w,v_w);
26    v_w = _mm256_hadd_epi32(v_w,v_w);
27    mm256_store_si256((__m256i *)&vv[0], v_w);
28    → dot = dot + vv[0] + vv[4];
29
30
31    //Epilogo
32    for(i = nvec1; i < n ; i++)
33        dot = dot + x[i]*y[i];
34    return dot;
35 }

```

```

71 static int idot_avx(int n, int *x, int *y)
72 {
73     int dot = 0.0;
74
75     int dir1 = (int) (&y[0]);
76     int npr11 = 16 - ((dir1>>2)%16);
77     int nvec1 = (((int) floor((double)((n-npr11)/8.0)))<<3)+ npr11;
78
79     int dir2 = (int) (&x[0]);
80     int npr12 = 16 - ((dir2>>2)%16);
81
82     if(npr11 == 16) npr11 = 0;
83     if(npr12 == 16) npr12 = 0;
84
85     if(npr11 == npr12) dot = alig_idot_avx(n,x,y, npr11, nvec1);
86     else dot = noalig_idot_avx(n,x,y, npr11, nvec1);
87
88     return dot;
89 }

```

```
6  int idot_omp(int n, int *x, int *y)
7  {
8      int i;
9      int dot = 0;
10     → #pragma omp parallel for reduction(+:dot)
11         for (i = 0; i < n; ++i)
12             dot = dot + x[i]*y[i];
13
14     return dot;
15 }
```

```
11 static int idot_omp_sse(int n, int *x, int *y)
12 {
13     int i,trozo,ini,fin;
14     int dot = 0;
15
16     trozo=ceil((float)n/(float)NTHR);
17
18     → #pragma omp parallel private(i,ini,fin) reduction(+:dot) firstprivate (n,trozo)
19         {
20             i = omp_get_thread_num();
21             ini=i*trozo;
22             fin=min(trozo*(i+1),n);
23             dot = idot_sse(fin-ini,&x[ini],&y[ini]);
24         }
25     return dot;
26 }
27
28 static int idot_omp_avx(int n, int *x, int *y)
29 {
30     int i,trozo,ini,fin;
31     int dot = 0;
32
33     trozo=ceil((float)n/(float)NTHR);
34
35     → #pragma omp parallel private(i,ini,fin) reduction(+:dot) firstprivate (n,trozo)
36         {
37             i = omp_get_thread_num();
38             ini=i*trozo;
39             fin=min(trozo*(i+1),n);
40             dot = idot_avx(fin-ini,&x[ini],&y[ini]);
41         }
42     return dot;
43 }
```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	2.678172	4.578947	0.004876	0.007452	0.007347
4	1000	2.840839	5.078683	0.013911	0.011174	0.012715
4	1500	2.904308	5.358306	0.016565	0.014856	0.016944
4	2000	2.910629	5.616327	0.019728	0.019234	0.023190
4	2500	2.968980	5.437823	0.023886	0.024905	0.025555
4	3000	2.938563	5.626200	0.028276	0.030605	0.030550
4	3500	3.014253	6.198046	0.043111	0.043081	0.039139
4	4000	2.966895	5.356414	0.042816	0.040954	0.049814
4	8000	2.337962	2.865260	0.085565	0.094981	0.096802
4	13000	2.474948	3.421586	0.126139	0.151489	0.144747
4	18000	3.556910	5.091463	0.280384	0.279115	0.300519
4	23000	1.901975	3.649450	0.215609	0.230556	0.263855
4	28000	2.612178	4.093095	0.321967	0.295617	0.307394
4	33000	2.537078	3.823049	0.271493	0.342379	0.361741
4	38000	1.922681	2.375499	0.418131	0.396188	0.406402
4	43000	1.944868	1.885319	0.467674	0.574243	0.473662
4	48000	1.834254	2.022666	0.394566	0.445105	0.443909
4	53000	1.798087	2.306491	0.529076	0.479317	0.517767
4	58000	2.072275	2.279272	0.485472	0.602938	0.570576
4	100000	2.275686	2.604727	0.724248	0.941585	1.137189
4	175000	2.389237	2.643411	1.021419	1.654111	1.761861
4	250000	2.211443	2.435043	1.052375	1.585860	1.867401
4	325000	2.353544	2.543927	1.117376	1.960882	2.141080
4	400000	2.284492	2.552040	1.268177	2.076811	2.403359
4	475000	2.231839	2.484557	1.141448	2.300205	2.459794
4	550000	2.345639	2.634523	1.950530	2.733404	2.905113
4	625000	2.072398	2.162582	1.461066	2.202355	2.933872
4	700000	2.095107	2.192732	1.688057	2.795026	3.151993
4	775000	1.955423	2.138399	1.673270	2.562332	3.307345
4	1600000	1.818035	1.899268	1.961728	3.156288	2.801137



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	2,633087	4,303625	0,005072	0,008863	0,008191
	2	1000	2,812401	7,911449	0,02324	0,027233	0,027204
	2	1500	1,711373	3,179354	0,011979	0,017757	0,026318
	2	2000	2,665708	4,904762	0,049785	0,056801	0,056963
	2	2500	2,959524	5,445241	0,0398	0,041803	0,038298
	2	3000	2,825836	4,822305	0,035962	0,059835	0,081711
	2	3500	4,078923	3,88124	0,059957	0,059834	0,041425
	2	4000	2,941926	2,547027	0,047394	0,045528	0,046455
L2	2	8000	2,577808	4,170089	0,148826	0,221717	0,230594
	2	13000	2,285935	2,957014	0,123507	0,173975	0,156001
	2	18000	2,649992	4,22185	0,290438	0,3639	0,37243
	2	23000	2,411855	3,572986	0,226633	0,22758	0,247951
	2	28000	1,843343	2,522408	0,32572	0,451236	0,417534
L3	2	33000	1,376495	1,816412	0,315059	0,443385	0,45965
	2	38000	2,123876	2,722004	0,571289	0,746061	0,613963
	2	43000	1,30451	1,679767	0,283554	0,510558	0,53551
	2	48000	2,001754	2,692927	0,560668	0,77079	0,660921
	2	53000	1,842652	1,969712	0,363965	0,555683	0,619987
	2	58000	2,456428	3,290845	0,731598	1,129299	1,164073
	2	100000	2,936669	3,164994	0,807279	0,977814	1,069777
	2	175000	1,92595	2,259098	0,780398	1,398799	1,485072
	2	250000	2,645874	2,936438	0,892679	1,832915	1,735718
	2	325000	2,146043	2,513681	0,966535	1,407948	1,534325
RAM	2	400000	2,167514	2,547526	1,097517	1,551356	1,765788
	2	475000	3,468679	3,825011	1,408417	2,681768	3,218244
	2	550000	2,038266	2,237118	0,906177	1,524619	1,475314
	2	625000	2,42654	2,289213	1,133884	1,994389	1,974552
	2	700000	2,733076	2,973985	1,715511	2,770622	2,736568
	2	775000	1,766812	1,841835	1,299925	1,735207	1,782621



	th	n	sse	paralelo	par-sse
L1	4	500	1,00974	0,00505	0,005248
	4	1000	1,085671	0,007374	0,007969
	4	1500	1,157186	0,016535	0,017302
	4	2000	1,138284	0,016512	0,016522
	4	2500	1,149584	0,024322	0,024069
	4	3000	1,161781	0,022766	0,023067
	4	3500	1,108297	0,039415	0,034332
	4	4000	1,17196	0,032793	0,032632
L2	4	8000	1,33096	0,082468	0,084373
	4	13000	1,328128	0,119463	0,110654
	4	18000	1,314323	0,206815	0,186678
	4	23000	1,305721	0,23332	0,245047
	4	28000	1,80465	0,340118	0,330584
	4	33000	1,314462	0,322619	0,360487
	4	38000	1,327582	0,359449	0,436122
	4	43000	1,342009	0,385835	0,395706
	4	48000	1,30341	0,370246	0,368122
	4	53000	1,309307	0,331388	0,421596
	4	58000	1,31043	0,463199	0,532938
	4	100000	1,524712	0,625656	0,640989
	4	175000	1,482168	0,758098	0,870338
	4	250000	1,685184	1,022786	1,089351
	4	325000	1,434318	1,11059	1,082161
	4	400000	1,637058	1,020642	1,243976
4	475000	1,540025	1,107317	1,117909	
4	550000	1,224711	0,812443	0,87567	
4	625000	1,277657	0,986816	1,196316	
4	700000	1,284336	0,801232	0,877823	
RAM	4	775000	1,138784	0,908434	1,290024
	4	1600000	0,782771	1,49527	1,307725



sdot

```

4  float sdot_seq(int n, float *x, float *y)
5  {
6      int i;
7      float dot = 0.0;
8      for (i = 0; i < n; ++i)
9          dot = dot + x[i]*y[i];
10
11     return dot;
12 }

```

```

10 static float align_sdot_sse(int n, float *x, float *y, int nprol, int nvecl)
11 {
12     int i,j;
13     float dot = 0.0;
14     float attribute ((aligned(64))) vv;
15     __m128 v_x, v_y, v_w;
16
17     //Prologo
18     for(i = 0; i < nprol ; i++)
19         dot = dot + x[i]*y[i];
20
21     //Cuerpo
22     v_w = _mm_setzero_ps();
23     for (i = nprol; i < nvecl; i+=4) {
24         v_x = _mm_load_ps(&x[i]);
25         v_y = _mm_load_ps(&y[i]);
26         v_y = _mm_mul_ps(v_y, v_x);
27         v_w = _mm_add_ps(v_y, v_w);
28     }
29     v_w = _mm_hadd_ps(v_w, v_w);
30     v_w = _mm_hadd_ps(v_w, v_w);
31     _mm_store_ss(&vv, v_w);
32     dot = dot + vv;
33
34
35     //Epilogo
36     for(i = nvecl; i < n ; i++)
37         dot = dot + x[i]*y[i];
38     return dot;
39 }

```

noalign_sdot_sse

```

v_x = _mm_loadu_ps(&x[i]);
v_y = _mm_load_ps(&y[i]);

```

```

6  static float align_sdot_avx(int n, float *x, float *y, int nprol, int nvecl)
7  {
8      int i,j;
9      float dot = 0.0;
10     float __attribute__((aligned(64))) vv[8];
11     __m256 v_x, v_y, v_w;
12
13     //Prologo
14     for(i = 0; i < nprol ; i++)
15         dot = dot + x[i]*y[i];
16
17     //Cuerpo
18     v_w = _mm256_setzero_ps();
19     for (i = nprol; i < nvecl; i+=8) {
20         v_x = _mm256_load_ps(&x[i]);
21         v_y = _mm256_load_ps(&y[i]);
22         v_y = _mm256_mul_ps(v_y, v_x);
23         v_w = _mm256_add_ps(v_y, v_w);
24     }
25     v_w = _mm256_hadd_ps(v_w, v_w);
26     v_w = _mm256_hadd_ps(v_w, v_w);
27     _mm256_store_ps(&vv[0], v_w);
28     dot = dot + vv[0] + vv[4];
29
30     //Epilogo
31     for(i = nvecl; i < n ; i++)
32         dot = dot + x[i]*y[i];
33     return dot;
34 }

```



```

6  static float align_sdot_fma(int n, float *x, float *y, int nprol, int nvecl)
7  {
8      int i,j;
9      float dot = 0.0;
10     float attribute ((aligned(64))) vv[8];
11     __m256 v_x, v_y, v_w;
12
13     //Prologo
14     for(i = 0; i < nprol ; i++)
15         dot = dot + x[i]*y[i];
16
17     //Cuerpo
18     v_w = mm256_setzero_ps();
19     for (i = nprol; i < nvecl; i+=8){
20         v_x = _mm256_load_ps(&x[i]);
21         v_y = mm256_load_ps(&y[i]);
22         v_w = _mm256_fmadd_ps(v_y,v_x,v_w);
23     }
24     v_w = mm256_hadd_ps(v_w,v_w);
25     v_w = _mm256_hadd_ps(v_w,v_w);
26     _mm256_store_ps(&vv[0], v_w);
27     dot = dot + vv[0] + vv[4];
28
29     //Epilogo
30     for(i = nvecl; i < n ; i++)
31         dot = dot + x[i]*y[i];
32     return dot;
33 }

```

```

46  static float sdot_omp_fma(int n, float *x, float *y)
47  {
48     int i,trozo,ini,fin;
49     float dot = 0.0;
50
51     trozo=ceil((float)n/(float)NTHR);
52
53     #pragma omp parallel private(i,ini,fin) reduction(+:dot) firstprivate (n,trozo)
54     {
55         i = omp_get_thread_num();
56         ini=i*trozo;
57         fin=min(trozo*(i+1),n);
58         dot = sdot_fma(fin-ini,&x[ini],&y[ini]);
59     }
60     return dot;
61 }

```

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	3.749826	6.854777	4.575680	0.011466	0.010820	0.013145	0.014639
4	1000	3.874544	7.241990	4.659649	0.025473	0.026476	0.026008	0.026360
4	1500	3.908666	7.582139	4.715474	0.030425	0.031625	0.014225	0.027749
4	2000	3.890055	7.628386	4.690873	0.041609	0.042104	0.043427	0.043442
4	2500	3.953109	7.738123	4.745863	0.050984	0.047375	0.052556	0.056215
4	3000	3.984016	7.723069	4.789671	0.062125	0.062766	0.033990	0.062957
4	3500	3.954867	7.758360	2.610089	0.071416	0.074950	0.076311	0.075392
4	4000	3.964713	7.827048	4.747915	0.090152	0.099324	0.100659	0.096966
4	8000	3.689465	4.735836	3.961448	0.138194	0.171371	0.168404	0.150570
4	13000	3.806163	5.597061	4.237695	0.276800	0.265615	0.259499	0.242771
4	18000	3.865862	6.147745	4.399558	0.319847	0.379504	0.333716	0.342740
4	23000	3.898318	6.236363	4.460596	0.422268	0.378009	0.408934	0.445812
4	28000	3.918687	6.704232	4.540413	0.563024	0.471970	0.511358	0.511857
4	33000	3.987993	6.824754	4.644954	0.520939	0.610686	0.599110	0.686645
4	38000	2.021901	3.967033	3.787984	0.703258	0.637601	0.647401	0.742857
4	43000	3.344064	3.669530	3.542579	0.677391	0.720048	0.807476	0.812203
4	48000	3.340824	3.656921	3.693579	0.722797	0.837597	1.019776	0.945117
4	53000	3.401520	3.839493	3.807104	0.777415	0.839145	1.023318	1.129407
4	58000	2.009828	3.707267	3.754838	0.786713	1.035213	0.971122	1.176605
4	100000	3.704470	4.357507	4.204728	1.072276	1.495619	1.630760	1.547845
4	175000	3.818545	4.525010	4.038125	1.220012	2.630402	2.819102	2.934816
4	250000	3.856818	4.730270	4.539656	1.927240	3.432564	3.724284	3.443460
4	325000	3.908076	4.905091	4.603240	1.487948	4.011391	4.405959	4.317255
4	400000	3.706119	4.815726	4.271470	1.637939	4.084992	4.488303	4.381242
4	475000	3.884584	4.846502	4.639358	2.662461	5.224535	4.991229	5.083981
4	550000	3.814582	4.443401	4.703497	1.736618	5.239362	6.064794	5.033692
4	625000	3.404177	4.685001	4.579550	1.798938	5.065277	5.030370	6.032582
4	700000	3.359411	4.616135	4.563188	2.174080	5.459618	6.586344	6.301290
4	775000	3.651821	4.034275	3.869359	2.016097	6.257515	5.888563	6.273144
4	1600000	3.255520	3.334554	3.397030	3.114893	5.773428	5.949997	5.756095



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma
L1	2	500	3,703576	6,895006	4,551986	0,008351	0,011559	0,011329	0,011562
	2	1000	3,88702	7,173414	4,519983	0,015667	0,021985	0,021688	0,021685
	2	1500	3,848729	7,204231	5,052211	0,027078	0,037574	0,037653	0,035998
	2	2000	3,810054	6,992422	4,718097	0,042699	0,036591	0,043184	0,043181
	2	2500	3,265257	7,862998	4,832674	0,068214	0,039701	0,068909	0,084233
	2	3000	3,587972	6,947965	5,778039	0,06513	0,086139	0,102761	0,07946
	2	3500	4,088515	8,07078	4,933996	0,092548	0,058652	0,080184	0,079339
	2	4000	4,364564	7,819502	6,847528	0,150795	0,168468	0,09908	0,132804
L2	2	8000	3,658521	4,585192	3,940024	0,135728	0,24448	0,211184	0,146433
	2	13000	3,806871	5,604976	4,193676	0,225798	0,384546	0,392776	0,223808
	2	18000	3,770447	5,739879	4,30504	0,302027	0,50892	0,478171	0,303624
	2	23000	3,505208	5,615447	4,863747	0,473549	0,685925	0,383263	0,48029
	2	28000	3,192666	5,046124	4,749336	0,52762	0,806718	0,611171	0,54333
L3	2	33000	3,343921	5,476364	5,173589	0,448764	0,894817	0,665927	0,668509
	2	38000	3,095782	4,362771	4,014048	0,697049	0,967883	0,895679	0,916404
	2	43000	3,491367	3,805586	3,797699	0,521227	0,670012	0,768836	0,764363
	2	48000	3,439324	3,745165	3,706064	0,531135	0,664745	1,01326	1,209356
	2	53000	3,489544	3,843914	3,794046	0,71754	0,864831	0,900352	1,400506
	2	58000	3,422237	4,23569	4,233195	0,096672	1,072922	0,86467	1,376106
	2	100000	3,711322	4,308512	4,202395	0,847861	1,403326	1,700117	1,676566
	2	175000	3,560403	4,663601	4,613285	0,612166	2,026389	3,20441	3,07969
	2	250000	3,95764	5,402162	5,253057	1,466306	2,828597	2,992763	2,859477
	2	325000	2,985437	3,49	3,018376	0,155261	0,276039	0,168194	0,453329
RAM	2	400000	3,905973	4,546508	4,314894	1,450687	3,201669	3,522918	3,52148
	2	475000	3,737167	4,126533	4,144995	1,491043	3,112608	3,296365	3,330798
	2	550000	3,952339	4,20234	4,174067	1,757781	3,577112	3,704599	2,955297
	2	625000	4,313347	4,474102	4,448863	1,951045	4,367351	4,242735	4,405558
	2	700000	3,874385	3,894909	3,88858	1,809697	4,115583	3,914875	3,78833
	2	775000	3,426472	3,495884	3,424371	1,605505	3,002871	2,950731	3,103217



	th	n	sse	paralelo	par-sse
L1	4	500	2,448819	0,007891	0,008042
	4	1000	2,977304	0,013789	0,014664
	4	1500	3,189776	0,016816	0,018097
	4	2000	3,305677	0,022772	0,023315
	4	2500	3,360239	0,033594	0,031443
	4	3000	3,428247	0,0354	0,033218
	4	3500	3,404192	0,048404	0,049051
	4	4000	3,411706	0,055422	0,055783
L2	4	8000	2,602636	0,103193	0,105483
	4	13000	2,644713	0,191381	0,167133
	4	18000	2,670931	0,295919	0,283032
	4	23000	3,82112	0,405757	0,359798
	4	28000	2,674287	0,371775	0,468091
	4	33000	2,688273	0,410285	0,475048
	4	38000	2,692199	0,41488	0,414778
	4	43000	2,656109	0,446146	0,462873
	4	48000	3,800082	0,656154	0,721263
	4	53000	2,68833	0,509658	0,53285
	4	58000	3,758382	0,773599	0,748303
	4	100000	3,158648	1,13398	0,963527
	4	175000	2,707533	0,965782	0,846845
	4	250000	2,207902	0,789113	1,0664
	4	325000	2,1902	0,879113	0,976400
	4	400000	2,136461	0,54058	0,763559
4	475000	1,12576	1,159691	0,922689	
4	550000	1,648551	1,653727	1,701746	
4	625000	1,189061	1,123675	1,126746	
4	700000	1,47412	1,193475	1,244679	
RAM	4	775000	1,176713	1,166073	1,081547
	4	1600000	0,924559	1,167209	1,240877



```

6  static float alig_sdot_avx(int n, float *x, float *y, int nprol, int nvecl)
7  {
8      int i,j;
9      float dot = 0.0;
10     float __attribute__((aligned(64))) vv[8];
11     __m256 v_x, v_y, v_w, v_x2, v_y2, v_w2;
12
13     //Prologo
14     for(i = 0; i < nprol ; i++)
15         dot = dot + x[i]*y[i];
16
17     //Cuerpo
18     v_w = _mm256_setzero_ps();
19     v_w2 = _mm256_setzero_ps();
20     for (i = nprol; i < nvecl; i+=16){
21         v_x = _mm256_load_ps(&x[i]);
22         v_y = _mm256_load_ps(&y[i]);
23         v_y = mm256_mul_ps(v_y, v_x);
24         v_w = _mm256_add_ps(v_y, v_w);
25         v_x2 = _mm256_load_ps(&x[i+8]);
26         v_y2 = mm256_load_ps(&y[i+8]);
27         v_y2 = _mm256_mul_ps(v_y2, v_x2);
28         v_w2 = _mm256_add_ps(v_y2, v_w2);
29     }
30     v_w = mm256_hadd_ps(v_w, v_w);
31     v_w = _mm256_hadd_ps(v_w, v_w);
32     mm256_store_ps(&vv[0], v_w);
33     dot = dot + vv[0] + vv[4];
34     v_w2 = _mm256_hadd_ps(v_w2, v_w2);
35     v_w2 = mm256_hadd_ps(v_w2, v_w2);
36     mm256_store_ps(&vv[0], v_w2);
37     dot = dot + vv[0] + vv[4];
38
39     //Epilogo
40     for(i = nvecl; i < n ; i++)
41         dot = dot + x[i]*y[i];
42     return dot;
43 }

```

Operation

```

FOR j := 0 to 3
  i := j*32
  dst[i+31:i] := (a[i+31:i] * b[i+31:i]) + c[i+31:i]
ENDFOR
dst[MAX:128] := 0

```

Performance

Architecture	Latency	Throughput
Haswell	5	0.5

```


84  static float sdot_avx(int n, float *x, float *y)
85  {
86     float dot = 0.0;
87
88     int dir1 = (int) (&y[0]);
89     int nprol1 = 16 - ((dir1>>2)%16);
90     int nvecl = (((int) floor((double) ((n-nprol1)/16.0)))<<4) + nprol1;
91
92     int dir2 = (int) (&x[0]);
93     int nprol2 = 16 - ((dir2>>2)%16);
94
95     if(nprol1 == 16) nprol1 = 0;
96     if(nprol2 == 16) nprol2 = 0;
97
98     if(nprol1 == nprol2) dot = alig_sdot_avx(n,x,y, nprol1, nvecl);
99     else dot = noalig_sdot_avx(n,x,y, nprol1, nvecl);
100
101     return dot;
102 }

```

Unroll 2

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	6.975357	9.402098	7.407713	0.013050	0.013232	0.012390	0.011107
4	1000	7.294441	11.808889	8.290172	0.021519	0.021379	0.022523	0.025280
4	1500	7.150450	12.062310	8.557412	0.033629	0.040164	0.044563	0.044394
4	2000	7.543929	13.557766	8.860319	0.035918	0.038173	0.049743	0.029578
4	2500	7.686972	14.294851	8.955518	0.056004	0.055852	0.058567	0.055167
4	3000	7.711257	14.121821	9.049757	0.064895	0.064773	0.062471	0.063137
4	3500	7.843571	14.021277	9.085180	0.064174	0.074512	0.071158	0.063187
4	4000	7.656852	13.054850	9.208306	0.081677	0.035056	0.101361	0.083986
4	8000	4.458212	4.905020	4.929997	0.150118	0.210259	0.168598	0.129960
4	13000	5.077769	6.018047	5.215089	0.242780	0.304569	0.248748	0.238167
4	18000	4.787712	6.535308	6.752273	0.078005	0.429466	0.479890	0.427088
4	23000	5.330159	6.732762	7.213917	0.337289	0.478107	0.468030	0.446964
4	28000	4.952505	7.407502	3.358247	0.501627	0.500534	0.564476	0.607714
4	33000	5.371036	6.680205	6.294261	0.517249	0.578621	0.632486	0.662708
4	38000	3.013319	3.904173	3.286263	0.717999	0.685737	0.800897	0.629353
4	43000	3.543216	3.744543	3.720629	0.652482	0.784794	0.785742	0.933300
4	48000	3.610511	3.728851	3.732549	0.681778	0.857000	0.931701	0.942420
4	53000	3.763905	3.838723	3.845668	0.770172	0.896416	0.980007	1.007580
4	58000	3.775497	3.935424	3.935653	0.809369	1.028870	1.001701	1.100347
4	100000	4.168670	4.351736	4.362235	0.978364	1.522102	1.669120	1.586682
4	175000	4.714685	4.729868	4.728793	1.648494	2.833829	2.941920	2.784751
4	250000	4.782503	4.736963	4.697257	1.543448	3.759649	3.886825	3.801908
4	325000	5.016304	4.939783	4.928751	2.011545	4.813803	4.757431	4.199963
4	400000	4.979020	4.900747	4.914758	2.559954	5.315845	4.650168	5.348572
4	475000	4.485335	4.165374	4.822820	1.618658	4.678510	4.943434	5.908393
4	550000	3.934246	4.343364	4.736727	2.316064	5.268561	6.775332	6.389688
4	625000	4.856369	4.806101	4.470946	2.024505	6.479399	6.901874	6.531379
4	700000	3.661425	4.403209	4.222132	2.059966	6.564797	6.461911	6.272176
4	775000	3.795581	4.630519	4.194451	2.558237	8.245809	7.121508	6.552400
4	1600000	3.324276	3.457111	3.522395	2.866464	5.773260	6.064766	6.612964



ddot


```

4  double ddot_seq(int n, double *x, double *y)
5  {
6      int i;
7      double dot = 0.0;
8      for (i = 0; i < n; ++i)
9          dot = dot + x[i]*y[i];
10
11     return dot;
12 }

```

```

10 static double align_ddot_sse(int n, double *x, double *y, int nprol, int nvecl)
11 {
12     int i,j;
13     double dot = 0.0;
14     double attribute ((aligned(64))) vv;
15     __m128d v_x, v_y, v_w;
16
17     //Prologo
18     for(i = 0; i < nprol ; i++)
19         dot = dot + x[i]*y[i];
20
21     //Cuerpo
22     v_w = _mm_setzero_pd();
23     for (i = nprol; i < nvecl; i+=2){
24         v_x = _mm_load_pd(&x[i]);
25         v_y = _mm_load_pd(&y[i]);
26         v_y = _mm_mul_pd(v_y, v_x);
27         v_w = _mm_add_pd(v_y, v_w);
28     }
29     v_w = _mm_hadd_pd(v_w, v_w);
30     _mm_store_sd(&vv, v_w);
31     dot = dot + vv;
32
33     //Epilogo
34     for(i = nvecl; i < n ; i++)
35         dot = dot + x[i]*y[i];
36     return dot;
37 }

```

```

v_w = _mm256_setzero_pd();
for (i = nprol; i < nvecl; i+=4){
    v_x = _mm256_load_pd(&x[i]);
    v_y = _mm256_load_pd(&y[i]);
    v_y = _mm256_mul_pd(v_y, v_x);
    v_w = _mm256_add_pd(v_y, v_w);
}
v_w = _mm256_hadd_pd(v_w, v_w);
_mm256_store_pd(&vv[0], v_w);
dot = dot + vv[0] + vv[2];

```

```

v_w = _mm256_setzero_pd();
for (i = nprol; i < nvecl; i+=4){
    v_x = _mm256_load_pd(&x[i]);
    v_y = _mm256_load_pd(&y[i]);
    v_w = _mm256_fmadd_pd(v_y, v_x, v_w);
}
v_w = _mm256_hadd_pd(v_w, v_w);
_mm256_store_pd(&vv[0], v_w);
dot = dot + vv[0] + vv[2];

```

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	1.968910	3.702201	2.373457	0.010714	0.011008	0.010847	0.011312
4	1000	1.970403	3.835794	2.364484	0.021148	0.021448	0.021704	0.021071
4	1500	1.982395	3.781930	2.401261	0.032766	0.031337	0.030732	0.026823
4	2000	1.992646	3.900701	2.394245	0.045633	0.049567	0.046596	0.049100
4	2500	1.772709	2.004827	1.866573	0.038027	0.058820	0.052691	0.052677
4	3000	2.186946	2.809724	2.294003	0.061960	0.092896	0.106411	0.114392
4	3500	1.839542	2.241234	2.009146	0.075064	0.074888	0.078672	0.076948
4	4000	1.845517	2.375470	1.978689	0.079274	0.090853	0.085152	0.083230
4	8000	1.928669	2.985286	2.178749	0.188078	0.176108	0.194281	0.163584
4	13000	2.128085	3.600791	2.458383	0.296808	0.328833	0.162306	0.273429
4	18000	1.818250	2.369147	2.124026	0.349782	0.388044	0.337092	0.358618
4	23000	1.627991	1.876031	1.814493	0.363515	0.416971	0.446987	0.470538
4	28000	1.707401	1.949873	1.229266	0.419388	0.457723	0.554589	0.576163
4	33000	1.716294	2.036002	1.984773	0.536200	0.519854	0.717926	0.581821
4	38000	1.799766	2.072212	1.996821	0.509863	0.626287	0.678098	0.699987
4	43000	1.787421	2.125691	2.086259	0.550553	0.712982	0.664868	0.689508
4	48000	1.871696	2.191956	2.067244	0.769180	0.884160	0.760237	0.839912
4	53000	1.860442	2.214542	2.118460	0.669029	0.842668	0.961837	0.893947
4	58000	1.865969	2.223178	2.139259	0.804193	0.934831	0.979707	0.963871
4	100000	1.920610	2.313411	2.243167	1.031909	1.496386	1.565643	1.539832
4	175000	1.964433	2.404089	2.310993	1.337374	2.330533	2.367853	2.454163
4	250000	1.964298	2.324546	2.229670	1.483971	2.305705	2.622649	2.453918
4	325000	2.061074	2.215585	2.470423	2.273002	3.018617	3.177914	3.435770
4	400000	1.951751	2.001483	2.308813	2.246603	3.422584	4.122029	3.466657
4	475000	1.689876	2.052739	1.939883	1.875989	2.879716	3.244212	3.280694
4	550000	1.696305	1.733701	1.812994	2.377888	2.788133	2.820101	2.725829
4	625000	1.725346	1.847644	1.806068	2.129516	3.130810	3.138370	3.249280
4	700000	1.768744	1.761791	1.771905	2.695206	3.111912	3.124019	3.185809
4	775000	1.699509	1.684133	1.688716	2.535662	3.066277	3.102362	3.087787
4	1600000	1.603546	1.656156	1.621519	2.494522	3.033465	2.991646	3.171382



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma
L1	2	500	1,971877	3,697945	2,380511	0,014464	0,016326	0,016502	0,016459
	2	1000	1,970947	3,791741	2,360071	0,016366	0,01907	0,027774	0,032397
	2	1500	1,977451	3,872408	2,382618	0,02429	0,029654	0,033199	0,034021
	2	2000	1,986652	3,905748	2,387213	0,033023	0,035454	0,041285	0,061818
L2	2	2500	1,772333	2,051785	1,962736	0,042171	0,041968	0,054167	0,054882
	2	3000	1,804053	2,093124	1,873718	0,076441	0,061887	0,054894	0,060256
	2	3500	1,898054	2,333192	2,009652	0,067884	0,066814	0,089375	0,106001
	2	4000	1,843577	2,380237	1,986529	0,107063	0,114006	0,067732	0,098437
	2	8000	1,924175	2,984749	2,174469	0,207275	0,226414	0,139136	0,217636
	2	13000	1,960717	3,277653	2,843698	0,351971	0,418252	0,431221	0,428812
L3	2	18000	1,86213	2,450528	2,137121	0,352919	0,259911	0,40319	0,398534
	2	23000	1,524914	1,855023	1,83927	0,455362	0,318	0,531181	0,518038
	2	28000	1,766961	1,956374	1,918652	0,436849	0,398293	0,51872	0,465359
	2	33000	1,461692	1,968018	1,996526	0,419196	0,463101	0,622109	0,657728
	2	38000	1,5102	2,103525	2,05125	0,51828	0,594619	0,588746	0,546539
	2	43000	1,725904	2,210475	2,162592	0,483498	0,682615	0,770538	0,766256
	2	48000	1,850948	2,142742	2,093461	0,565497	0,642241	0,810739	0,801158
	2	53000	1,776724	2,208851	2,119751	0,553577	0,77041	0,809035	0,769452
	2	58000	1,781162	2,268903	2,184671	0,61776	0,684664	0,927857	0,915979
	2	100000	1,774025	2,490903	2,615599	0,999287	1,229736	1,336183	1,28981
	2	175000	1,948117	2,396128	2,348488	1,3581	1,981399	1,956028	1,985412
RAM	2	250000	1,914353	1,853692	1,769929	1,144114	1,551522	1,781846	1,815311
	2	325000	1,823792	1,856158	1,91162	1,501628	2,003448	1,732277	1,46519
	2	400000	1,670543	1,741641	1,77387	1,372667	1,693862	1,719228	1,724487
	2	475000	1,672182	1,664469	1,687976	1,396877	1,674905	1,669896	1,72711
	2	550000	1,695286	1,671845	1,62464	1,109725	1,598513	1,750901	1,686422
	2	625000	1,588392	1,606645	1,59318	1,415129	1,702758	1,484264	1,754478
	2	700000	1,702669	1,64064	1,602088	1,512131	1,878162	2,04109	1,941534
	2	775000	1,816956	1,820073	1,828134	1,902835	2,143742	2,156425	2,046602



	th	n	sse	paralelo	par-sse
L1	4	500	1,549613	0,006997	0,007083
	4	1000	1,763416	0,010813	0,011692
	4	1500	1,817964	0,017056	0,01752
	4	2000	1,822156	0,027123	0,029664
L2	4	2500	1,35443	0,036429	0,036559
	4	3000	1,354945	0,044149	0,045503
	4	3500	1,357279	0,046552	0,051167
	4	4000	1,376667	0,055358	0,060139
	4	8000	1,355794	0,111773	0,107007
	4	13000	1,353312	0,135708	0,139474
	4	18000	1,347596	0,200457	0,258589
	4	23000	1,367454	0,273644	0,276985
	4	28000	1,353279	0,267646	0,276832
	4	33000	1,355096	0,346471	0,405007
	4	38000	1,348172	0,439329	0,433272
	4	43000	1,334949	0,352789	0,350324
	4	48000	1,349442	0,433018	0,470567
	4	53000	1,359942	0,241651	0,303163
	4	58000	1,356332	0,330632	0,352439
	4	100000	1,573393	0,591021	0,657691
4	175000	1,655865	0,567836	0,728849	
4	250000	1,340082	0,688419	0,603733	
4	325000	0,648481	0,559934	0,538974	
RAM	4	400000	1,055798	1,255174	1,217763
	4	475000	0,968376	1,544919	1,456739
	4	550000	0,962442	1,112187	1,11778
	4	625000	0,973237	1,316941	1,278931
	4	700000	0,92837	1,231203	1,269981
	4	775000	0,969697	1,102685	1,507033



dsdot

```

4 double dsdot_seq(int n, float *x, float *y)
5 {
6     int i;
7     double dot = 0.0;
8     for (i = 0; i < n; ++i)
9         dot = dot + (double)x[i]* (double)y[i];
10
11     return dot;
12 }

```

```

10 static double align_dsdot_sse(int n, float *x, float *y, int nprol, int nvecl)
11 {
12     int i,j;
13     double dot = 0.0;
14     double attribute__((aligned(64))) vv;
15     __m128 v_x, v_y;
16     __m128d v_xn, v_yn, v_w;
17
18     //Prologo
19     for(i = 0; i < nprol ; i++)
20         dot = dot + (double)x[i]* (double)y[i];
21
22     //Cuerpo
23     v_w = mm setzero pd();
24     for (i = nprol; i < nvecl; i+=2){
25         v_x = _mm_load_ps(&x[i]);
26         v_xn = mm cvtps pd(v x);
27         v_y = _mm_load_ps(&y[i]);
28         v_yn = _mm_cvtps_pd(v_y);
29         v_yn = mm mul pd(v_yn,v_xn);
30         v_w = _mm_add_pd(v_yn,v_w);
31     }
32     v_w = mm hadd pd(v_w,v_w);
33     mm store sd(&vv, v_w);
34     dot = dot + vv;
35
36     //Epilogo
37     for(i = nvecl; i < n ; i++)
38         dot = dot + (double)x[i]* (double)y[i];
39     return dot;
40 }

```

```

v_w = _mm256_setzero_pd();
for (i = nprol; i < nvecl; i+=4){
    v_x = _mm_load_ps(&x[i]);
    v_xn = _mm256_cvtps_pd(v_x);
    v_y = _mm_load_ps(&y[i]);
    v_yn = mm256 cvtps pd(v_y);
    v_yn = _mm256_mul_pd(v_yn,v_xn);
    v_w = _mm256_add_pd(v_yn,v_w);
}
v_w = _mm256_hadd_pd(v_w,v_w);
_mm256_store_pd(&vv[0], v_w);
dot = dot + vv[0] + vv[2];

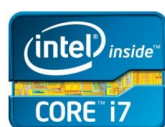
```

```

v_w = _mm256_setzero_pd();
for (i = nprol; i < nvecl; i+=4){
    v_x = _mm_load_ps(&x[i]);
    v_xn = _mm256_cvtps_pd(v_x);
    v_y = _mm_load_ps(&y[i]);
    v_yn = _mm256_cvtps_pd(v_y);
    v_yn = _mm256_fmadd_pd(v_yn,v_xn,v_w);
}
v_w = _mm256_hadd_pd(v_w,v_w);
_mm256_store_pd(&vv[0], v_w);
dot = dot + vv[0] + vv[2];

```

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	1.839822	3.446577	2.338108	0.010486	0.010433	0.010280	0.009750
4	1000	1.486400	4.241984	2.796840	0.021158	0.023814	0.022512	0.028321
4	1500	1.855294	3.624374	2.379749	0.029405	0.028730	0.031086	0.031020
4	2000	1.730755	3.257902	2.933800	0.055272	0.061323	0.072908	0.076052
4	2500	1.811082	3.538543	2.394290	0.011447	0.072213	0.065450	0.061309
4	3000	1.864540	3.674165	2.395917	0.073088	0.069629	0.071664	0.058040
4	3500	1.860157	3.674819	2.391854	0.068137	0.075343	0.072887	0.081332
4	4000	1.861127	3.700535	2.395898	0.081316	0.085298	0.083858	0.075262
4	8000	1.863892	3.576995	2.395305	0.117046	0.219140	0.221539	0.170339
4	13000	1.805966	3.678040	2.423714	0.170851	0.216682	0.304021	0.291089
4	18000	1.820889	2.641074	2.396436	0.368309	0.354899	0.340024	0.342900
4	23000	1.868124	3.444119	2.397722	0.431691	0.430627	0.464588	0.488541
4	28000	1.861527	3.659301	2.397984	0.511911	0.455376	0.562610	0.525881
4	33000	1.865129	3.671677	2.396050	0.456930	0.498692	0.642426	0.704468
4	38000	1.866256	3.217330	2.396988	0.606007	0.574670	0.730209	0.757064
4	43000	2.009132	3.339080	2.599389	0.710635	0.695816	0.889470	0.841889
4	48000	1.862879	3.169011	2.393014	0.728361	0.773089	0.941731	0.926298
4	53000	1.828835	3.193800	2.395484	0.777172	0.856457	0.842015	1.032147
4	58000	1.964696	3.460395	2.520744	0.724310	0.836321	0.950944	1.147737
4	100000	1.836642	3.573796	2.476155	1.061651	1.398575	1.756176	1.602850
4	175000	1.898440	3.619424	2.439984	1.312348	1.844741	2.611453	2.092905
4	250000	1.864405	3.596413	2.398090	1.666064	2.222779	2.980054	2.489183
4	325000	1.859421	3.589384	2.208962	1.494773	2.369832	3.281280	2.876310
4	400000	1.848862	3.645363	2.312483	2.008832	2.497197	3.749938	3.685606
4	475000	1.856659	3.420158	2.456433	2.024100	2.672365	5.748609	4.460500
4	550000	1.832512	3.741777	2.430222	1.832248	3.482149	5.161019	4.583187
4	625000	1.805938	3.703814	2.463001	1.887294	4.177160	5.761162	5.060898
4	700000	1.780622	3.506061	2.377788	2.223352	4.326687	5.801788	4.902051
4	775000	1.743769	3.103783	2.404330	2.718161	3.740900	5.716250	5.531566
4	1600000	1.870130	3.317453	2.483352	3.285081	5.029184	5.669926	5.994092



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma
L1	2	500	1,823748	3,48932	2,354148	0,008395	0,011685	0,011423	0,011049
	2	1000	1,839073	3,601491	2,379758	0,017081	0,021591	0,021627	0,021615
	2	1500	1,848143	3,621634	2,369718	0,024424	0,033013	0,033388	0,032863
	2	2000	1,845536	3,626438	2,381242	0,03274	0,040945	0,043024	0,042942
	2	2500	1,845116	3,662826	2,388396	0,041401	0,076948	0,077124	0,049531
	2	3000	1,850652	3,665084	2,393043	0,06253	0,057071	0,096543	0,083387
	2	3500	1,860108	3,666964	2,398316	0,08246	0,063088	0,074427	0,074048
	2	4000	1,853554	3,685737	2,398295	0,101269	0,123217	0,102903	0,125277
L2	2	8000	1,853352	3,577321	2,393821	0,125806	0,160573	0,163431	0,162114
	2	13000	2,047454	3,993783	2,63508	0,215967	0,275317	0,283839	0,278929
	2	18000	1,853495	3,655603	2,395326	0,261375	0,331163	0,384416	0,430096
	2	23000	1,777156	4,715249	3,08572	0,548764	0,515675	0,517376	0,537163
	2	28000	1,853463	3,691398	2,398867	0,36979	0,476221	0,626492	0,687215
L3	2	33000	1,839795	3,655703	2,396012	0,391437	0,69326	0,833769	0,752328
	2	38000	2,228415	4,229618	3,108663	0,562903	0,884836	1,230475	1,024702
	2	43000	1,795136	3,126258	2,391322	0,480645	0,711682	0,953746	0,880232
	2	48000	1,885321	3,282137	2,4313	0,52173	0,765666	1,190948	0,808534
	2	53000	1,85072	3,287365	2,394885	0,568169	0,832287	0,902922	0,875595
	2	58000	1,906912	3,441197	2,461088	0,592652	0,900696	1,058068	1,063377
	2	100000	1,547147	3,053638	2,924231	0,95234	1,528788	1,949251	1,878841
	2	175000	1,833089	3,576511	2,402329	1,092412	1,465541	1,775433	1,941039
	2	250000	1,86069	3,648846	2,424078	1,274928	1,805846	2,310417	2,04091
	2	325000	1,268246	2,52217	2,422387	1,380105	1,942431	2,677636	2,477496
RAM	2	400000	1,875257	3,495086	2,429811	1,444417	2,078924	2,861561	2,892159
	2	475000	1,868556	3,459773	2,37443	1,567397	2,220478	3,159137	2,939853
	2	550000	1,848987	3,484133	2,444046	1,533445	2,287266	3,094726	2,858672
	2	625000	1,549368	3,701001	2,665508	1,707236	2,59058	3,555502	3,257665
	2	700000	1,805667	2,84104	2,357919	1,538955	2,32721	3,286813	2,910979
	2	775000	1,838239	3,04665	2,400406	1,581147	2,496437	3,392027	3,009636



	th	n	sse	paralelo	par-sse
L1	4	500	1,027998	0,007004	0,006896
	4	1000	1,03666	0,012034	0,012286
	4	1500	1,086661	0,021646	0,019846
	4	2000	1,043701	0,026246	0,0256
	4	2500	1,044408	0,033256	0,034124
	4	3000	1,044321	0,03701	0,039494
	4	3500	1,024981	0,039477	0,036968
	4	4000	1,045574	0,055642	0,056302
L2	4	8000	1,047762	0,087254	0,089095
	4	13000	1,049058	0,140917	0,155797
	4	18000	1,067911	0,220242	0,225405
	4	23000	1,042	0,32676	0,32111
	4	28000	1,04942	0,321676	0,342111
	4	33000	1,048362	0,344564	0,354105
	4	38000	1,485642	0,592627	0,616607
	4	43000	1,045663	0,410331	0,418225
	4	48000	1,047962	0,489731	0,50802
	4	53000	1,048395	0,541232	0,51522
	4	58000	1,048036	0,515988	0,560666
	4	100000	1,49188	1,185113	1,213356
	4	175000	1,357615	1,178479	1,613163
	4	250000	1,045516	0,963869	1,319025
	4	325000	1,082141	1,435511	1,823736
	4	400000	1,043493	1,044245	0,963431
	4	475000	1,100958	1,205791	0,830694
	4	550000	1,005065	0,855858	0,734799
4	625000	1,060312	0,963663	1,450171	
4	700000	0,89459	1,305445	1,256018	
RAM	4	775000	0,923682	1,658264	1,345575



sdsdot

```

4  float sdsdot_seq(int n, float *x, float *y)
5  {
6      int i;
7      double dot = 0.0;
8      float result = 0.0;
9      for (i = 0; i < n; ++i)
10         dot = dot + (double)x[i]* (double)y[i];
11
12     result = dot;
13     return result;
14 }

```

```

10 static double alig_sdsdot_sse(int n, float *x, float *y, int nprol, int nvecl)
11 {
12     int i,j;
13     double dot = 0.0;
14     double __attribute__((aligned(64))) vv;
15     m128 v_x, v_y;
16     __m128d v_xn, v_yn, v_w;
17
18     //Prologo
19     for(i = 0; i < nprol ; i++)
20         dot = dot + (double)x[i]* (double)y[i];
21
22     //Cuerpo
23     v_w = _mm_setzero_pd();
24     for (i = nprol; i < nvecl; i+=2){
25         v_x = _mm_load_ps(&x[i]);
26         v_xn = _mm_cvtps_pd(v_x);
27         v_y = _mm_load_ps(&y[i]);
28         v_yn = _mm_cvtps_pd(v_y);
29         v_yn = _mm_mul_pd(v_yn,v_xn);
30         v_w = _mm_add_pd(v_yn,v_w);
31     }
32     v_w = _mm_hadd_pd(v_w,v_w);
33     mm_store_sd(&vv, v_w);
34     dot = dot + vv;
35
36     //Epilogo
37     for(i = nvecl; i < n ; i++)
38         dot = dot + (double)x[i]* (double)y[i];
39     return dot;
40 }

```

```

double dot = 0.0;
float result = 0.0;

```

```
dot = alig_sdsdot_sse(n,x,y, nprol1, nvecl);
```

```

result = dot;
return result;

```

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	1.817291	3.507823	2.360088	0.011186	0.010623	0.011380	0.013216
4	1000	1.832443	3.656691	2.371709	0.016383	0.030995	0.031254	0.029866
4	1500	1.812721	3.611010	2.387786	0.043846	0.039594	0.045545	0.045378
4	2000	1.849589	3.641786	2.382542	0.043719	0.041482	0.044753	0.039978
4	2500	1.855646	3.698809	2.395045	0.060387	0.051630	0.059973	0.058252
4	3000	1.855986	3.687303	2.392103	0.058261	0.064714	0.064896	0.062889
4	3500	1.856452	3.685994	2.382771	0.066289	0.067861	0.070727	0.066406
4	4000	1.859539	3.724010	2.392707	0.084440	0.090545	0.087215	0.074031
4	8000	1.798355	3.459381	2.430063	0.180336	0.149432	0.157065	0.195093
4	13000	1.812849	3.662469	2.397807	0.271196	0.243763	0.239915	0.260703
4	18000	1.863592	3.682277	2.398045	0.323899	0.320075	0.376547	0.350561
4	23000	1.789482	3.692196	2.399391	0.422810	0.388131	0.408799	0.397675
4	28000	1.865541	3.701614	2.400116	0.398412	0.449086	0.612058	0.632615
4	33000	1.806134	3.624650	2.396498	0.492755	0.542840	0.669016	0.729908
4	38000	1.851265	3.360293	2.425932	0.537724	0.678172	0.771527	0.464162
4	43000	1.762795	2.998717	2.387543	0.710188	0.761543	0.796760	0.794415
4	48000	1.269738	3.233410	2.408615	0.675304	0.713858	0.907966	1.000541
4	53000	1.878516	3.299688	2.420967	0.744876	0.882597	0.943986	0.963084
4	58000	2.050136	3.604055	2.488694	0.812648	0.948941	1.169226	1.047957
4	100000	1.923189	3.383881	2.480264	0.997703	1.419243	2.045959	1.674633
4	175000	1.888061	3.477499	2.560663	1.723518	2.158033	2.925926	2.569653
4	250000	1.831926	3.633826	2.407749	1.445721	2.055038	2.924481	2.522758
4	325000	1.797153	3.625595	2.382166	1.421534	2.417236	3.483882	2.824610
4	400000	1.919272	3.619399	2.254466	1.588394	2.006244	3.962694	3.774836
4	475000	1.805652	2.714145	2.429857	1.621858	2.685727	4.484254	3.401100
4	550000	1.847308	3.686330	2.428672	2.592897	4.343182	4.836589	3.508857
4	625000	1.675976	3.695777	2.417377	1.841399	3.930411	4.510075	4.632749
4	700000	1.718418	3.391164	2.449372	1.805496	4.205569	6.100076	4.946250
4	775000	1.882766	3.592836	2.353281	2.801151	3.808712	5.967685	5.212784
4	1600000	1.986083	3.407396	2.660352	3.368884	4.601955	6.371297	6.328000



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma
L1	2	500	1,845601	3,49158	2,333766	0,008853	0,01333	0,012586	0,01604
	2	1000	1,84574	3,661618	2,368515	0,018006	0,032867	0,033119	0,033016
	2	1500	1,769672	3,584375	2,406775	0,026054	0,039725	0,049639	0,049264
	2	2000	1,853646	3,666435	2,382542	0,034503	0,065401	0,061853	0,065263
	2	2500	1,434528	3,357394	2,38951	0,057417	0,045014	0,082093	0,068395
	2	3000	1,996584	3,250479	2,560701	0,066544	0,05369	0,069723	0,069113
	2	3500	1,905015	3,786155	2,465894	0,096206	0,111883	0,114651	0,099445
	2	4000	1,872829	3,731899	2,389515	0,095612	0,095356	0,107102	0,069681
L2	2	8000	2,100885	3,902557	2,670512	0,173769	0,263437	0,247422	0,164519
	2	13000	1,880811	3,687547	2,412473	0,212627	0,251236	0,416945	0,351211
	2	18000	1,428972	2,839113	2,734053	0,418911	0,326642	0,45122	0,538931
	2	23000	1,791749	2,972974	2,439579	0,347504	0,471032	0,394492	0,437521
	2	28000	1,776826	3,010373	2,367462	0,440493	0,573171	0,523571	0,472337
L3	2	33000	1,504687	2,541587	2,566245	0,570058	0,723549	0,497033	0,767659
	2	38000	1,417744	2,385111	2,365471	0,584365	0,662936	0,670903	0,568254
	2	43000	1,812829	3,555763	2,724642	0,768043	0,676779	0,822364	0,769426
	2	48000	1,405852	2,599099	2,603565	0,738687	0,917717	0,873526	0,830603
	2	53000	1,71857	3,114911	2,408503	0,665545	0,849756	0,833141	0,817569
	2	58000	1,627807	3,103378	3,041508	0,796705	0,937261	1,099028	1,264455
	2	100000	1,840392	3,627667	2,505748	0,966818	1,30765	1,548253	1,355387
	2	175000	1,618055	3,633222	2,614055	1,297052	1,86868	2,02857	1,738969
	2	250000	1,75978	3,711174	2,411325	1,2929	1,839742	2,370305	2,06596
	2	325000	1,991518	3,829377	2,645585	1,560509	2,495096	3,833198	2,878089
RAM	2	400000	1,876916	3,637296	2,45417	1,567001	2,629724	3,939552	2,537323
	2	475000	1,909013	3,62243	2,514937	1,569538	2,423397	3,434826	3,237387
	2	550000	1,9658	3,409808	2,51962	1,505919	2,460194	3,138243	3,001628
	2	625000	1,996708	3,782512	2,627	1,809594	2,243347	3,096454	3,313522
	2	700000	1,79812	3,31884	2,44059	1,53386	2,45244	3,275759	3,24613
	2	775000	1,837173	3,27292	2,403474	1,646652	2,616855	2,987293	2,89096



	th	n	sse	paralelo	par-sse
L1	4	500	1,031323	0,006988	0,007156
	4	1000	1,040462	0,011074	0,010937
	4	1500	1,043966	0,01785	0,016834
	4	2000	1,044669	0,027647	0,027112
	4	2500	1,045574	0,034637	0,033603
	4	3000	1,045458	0,043428	0,038225
	4	3500	1,046014	0,046773	0,050503
	4	4000	1,046697	0,049626	0,046129
L2	4	8000	1,246277	0,104998	0,110028
	4	13000	1,047864	0,196908	0,18188
	4	18000	1,04873	0,22162	0,251242
	4	23000	1,04349	0,3174	0,3096
	4	28000	1,045349	0,313174	0,326096
	4	33000	1,047732	0,369393	0,326813
	4	38000	1,048631	0,44611	0,456456
	4	43000	1,046858	0,452179	0,4445
	4	48000	1,048526	0,458314	0,375421
	4	53000	1,265515	0,630512	0,639481
	4	58000	1,046798	0,613366	0,801015
	4	100000	1,048507	0,583132	0,715752
	4	175000	1,248326	1,663361	1,940265
	4	250000	1,047951	1,020825	1,229846
	4	325000	1,054068	1,754542	2,216632
	4	400000	0,932714	0,683894	1,111186
4	475000	1,193932	1,471124	1,282658	
4	550000	0,738687	1,275625	1,278272	
4	625000	0,7867	1,35625	1,118272	
4	700000	0,972138	1,023711	1,080485	
RAM	4	775000	0,87667	1,273698	1,296084



amax**zi-amax**

```

4  int zi_amax_seq(int n, int *x)
5  {
6      int max;
7      int i, iamax;
8
9      max = abs(x[0]);
10     iamax = 0;
11     for(i=1; i<n; i++){
12         if (abs(x[i]) > max){
13             iamax = i;
14             max = abs(x[i]);
15         }
16     }
17     return iamax;
18 }

```

```

7  static int zi_amax_omp_proc(int desde, int hasta, int *x)
8  {
9      int i;
10     int max = abs(x[desde]);
11     int iamax = 0;
12     for(i=desde; i<hasta; i++){
13         if (abs(x[i]) > max){
14             iamax = i;
15             max = abs(x[i]);
16         }
17     }
18     return iamax;
19 }
20 static int zi_amax_omp(int n, int *x)
21 {
22     int i, ini, fin, trozo, ind, indv;
23     int maximo;
24     int *iamax = malloc(NTHR*sizeof(int));
25     trozo = ceil((float)n/(float)NTHR);
26
27     #pragma omp parallel private (i, ini, fin) firstprivate (n, trozo)
28     {
29         i = omp_get_thread_num();
30         ini = i*trozo;
31         fin = min(trozo*(i+1), n);
32         iamax[i] = zi_amax_omp_proc(ini, fin, x);
33     }
34
35     ind = iamax[0];
36     maximo = abs(x[ind]);
37     for (i=1; i<NTHR; i++){
38         indv = iamax[i];
39         if (maximo < abs(x[indv])){
40             maximo = abs(x[indv]);
41             ind = iamax[i];
42         }
43     }
44     free(iamax);
45     return ind;
46 }

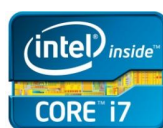
```

```

8 static int align_zi_amax_sse(int n, int *x, int nprol, int nvecl)
9 {
10     int max = 0;
11     int i,j,iamax,ind;
12     int __attribute__((aligned(64))) vindx[4];
13     __m128i v_x, ind_ant, var_ant, var_max, ind_new, mask, indand, indnand;
14
15     max = abs(x[0]);
16     iamax = 0;
17
18     //Prologo
19     for(i=0;i<nprol;i++){
20         if (abs(x[i]) > max){
21             iamax = i;
22             max = abs(x[i]);
23         }
24     }
25     ind_ant = _mm_set1_epi32(iamax);
26     var_ant = _mm_set1_epi32(max);
27     ind_new = _mm_setr_epi32(nprol, nprol+1, nprol+2, nprol+3);
28     //Cuerpo
29     for (i = nprol; i < nvecl; i+=4){
30         v_x = _mm_load_si128((__m128i *)&x[i]);
31         v_x = _mm_abs_epi32(v_x);
32         var_max = _mm_max_epi32(v_x,var_ant);
33
34         mask = _mm_cmpeq_epi32(var_max, var_ant);
35
36         indand = _mm_and_si128(mask,ind_ant);
37         indnand = _mm_andnot_si128(mask,ind_new);    not(mask) & ind_new
38
39         ind_ant = _mm_add_epi32(indand,indnand);
40         ind_new = _mm_add_epi32(ind_new, _mm_set1_epi32(4));
41         var_ant = var_max;
42     }
43     _mm_store_si128((__m128i *)&vindx[0],ind_ant);
44
45     for(i=0;i<4;i++){
46         ind = (int) vindx[i];
47         if(abs(x[ind]) > max){
48             max = abs(x[ind]);
49             iamax = ind;
50         }else if(abs(x[ind])== max && ind < iamax){
51             max = abs(x[ind]);
52             iamax = ind;
53         }
54     }
55
56     //Epilogo
57     for(i = nvecl; i < n ; i++){
58         if (abs(x[i]) > max){
59             iamax = i;
60             max = abs(x[i]);
61         }
62     }
63     return iamax;
64 }

```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	1.804724	2.942678	0.011614	0.015006	0.015488
4	1000	1.798319	3.082093	0.026588	0.008551	0.021857
4	1500	1.803209	3.101695	0.050233	0.055177	0.055531
4	2000	1.798589	3.148697	0.049963	0.053892	0.061656
4	2500	1.943230	3.480578	0.076270	0.082362	0.072099
4	3000	1.806921	3.175737	0.072186	0.077963	0.080214
4	3500	1.804157	3.168635	0.081606	0.090095	0.083938
4	4000	1.801335	3.168739	0.112949	0.120008	0.108867
4	8000	1.803545	3.194824	0.184030	0.201678	0.188782
4	13000	1.945460	3.442274	0.313960	0.328247	0.291554
4	18000	1.944288	3.448036	0.484110	0.452696	0.406478
4	23000	1.808985	3.209130	0.497228	0.527080	0.496004
4	28000	1.936856	3.455606	0.495173	0.637881	0.818949
4	33000	1.814662	3.222201	0.653456	0.601954	0.771301
4	38000	1.845308	3.366881	0.728576	0.787725	0.883115
4	43000	1.955988	3.481974	0.769536	0.952978	1.084493
4	48000	1.611590	3.203741	0.668785	0.865304	0.947871
4	53000	1.955667	3.600319	0.857371	1.086808	1.302400
4	58000	1.744927	3.215047	0.826386	1.152002	1.231619
4	100000	1.788881	3.232998	1.229238	1.768720	1.837037
4	175000	1.743491	3.231610	1.273011	1.986633	2.821914
4	250000	1.817186	3.296946	1.425376	2.660063	3.487832
4	325000	1.815021	3.226257	1.665474	2.998431	4.368350
4	400000	1.934859	3.467516	2.483907	3.951367	5.623602
4	475000	1.789633	3.128920	1.582006	3.118642	4.392389
4	550000	1.865113	3.338794	1.763547	2.991852	4.556284
4	625000	1.841897	3.308876	2.404385	2.994688	4.674706
4	700000	1.836102	3.259258	1.843316	3.329157	7.261656
4	775000	1.828229	3.255727	1.818258	3.394058	5.162681
4	1600000	1.823688	3.282503	3.211964	5.533538	8.432765



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	1,856131	3,015494	0,019745	0,020556	0,020612
	2	1000	1,792753	3,083174	0,019556	0,03019	0,03391
	2	1500	2,000375	3,094975	0,034408	0,041565	0,043458
	2	2000	2,941444	5,343545	0,105499	0,129665	0,142954
	2	2500	2,657724	4,933597	0,07642	0,099934	0,100658
	2	3000	1,962077	3,446059	0,061388	0,103505	0,126924
	2	3500	1,958034	3,480537	0,108403	0,125094	0,082797
	2	4000	2,745311	4,919038	0,155124	0,160147	0,217778
	2	8000	1,816276	3,216983	0,160345	0,18556	0,299312
L2	2	13000	2,652838	4,699871	0,455458	0,395415	0,562062
	2	18000	2,182118	5,31972	0,515899	0,658254	1,12471
	2	23000	2,093683	3,581964	0,481661	0,823385	0,755686
	2	28000	1,948143	3,423203	0,552805	0,943043	0,647849
	2	33000	2,318108	4,585771	0,545781	1,122359	0,984642
	2	38000	2,009326	3,769679	0,606054	1,150591	1,125695
	2	43000	1,984516	3,721637	0,619108	1,222037	1,43239
	2	48000	2,647	4,703288	0,84822	1,532934	1,379328
	2	53000	2,337719	4,28218	0,815494	1,597663	1,913223
	2	58000	1,991461	3,729746	0,610909	1,101904	1,524653
L3	2	100000	2,013353	3,635439	0,882467	1,332086	1,990515
	2	175000	2,138813	3,73367	1,296607	1,917161	2,862195
	2	250000	1,869511	3,34934	1,298893	2,02305	2,646343
	2	325000	2,379356	4,285283	1,776424	2,904512	4,129433
	2	400000	1,813735	3,248392	1,409659	2,355489	3,265614
	2	475000	1,808967	3,269147	1,467198	2,519067	3,583873
	2	550000	2,037361	3,667961	1,714039	2,903171	4,294772
	2	625000	2,027544	3,635824	1,733116	2,961796	4,333004
	2	700000	1,805222	3,225599	0,986236	2,749612	4,063061
2	775000	2,082357	3,712469	1,708944	3,153674	4,762644	



	th	n	sse	paralelo	par-sse
L1	4	500	2,68905	0,010498	0,009863
	4	1000	2,725867	0,025753	0,024788
	4	1500	2,722163	0,028369	0,028702
	4	2000	2,751839	0,036554	0,038171
	4	2500	2,757715	0,042375	0,045906
	4	3000	2,755474	0,05328	0,054895
	4	3500	2,76363	0,053298	0,0828
	4	4000	2,767541	0,092204	0,092937
	4	8000	2,761606	0,18891	0,181085
L2	4	13000	2,762563	0,298516	0,306471
	4	18000	2,76572	0,334511	0,367246
	4	23000	2,766085	0,481077	0,599917
	4	28000	2,770445	0,489843	0,599011
	4	33000	2,771251	0,573618	0,753792
	4	38000	2,786439	0,650613	0,769582
	4	43000	2,76774	0,712193	0,879747
	4	48000	2,782937	0,751367	0,938125
	4	53000	2,768449	0,892519	1,080345
	4	58000	3,510347	1,0263	1,179891
	4	100000	2,759438	1,178849	1,635333
	4	175000	3,280689	2,153512	3,199234
	4	250000	3,584103	2,410683	4,402169
	4	325000	3,521555	2,147456	4,15279
	4	400000	2,774563	1,777709	3,31862
	4	475000	2,775267	1,853244	3,977149
	4	550000	2,72118	2,294669	3,766313
4	625000	2,954531	2,122042	4,333136	
4	700000	2,773278	1,931337	4,320573	
4	775000	2,861626	1,795067	4,390247	



si-amax

```

4  int si_amax_seq(int n, float *x)
5  {
6      float max;
7      int i, iamax;
8
9      max = fabsf(x[0]);
10     iamax = 0;
11     for (i=1; i<n; i++){
12         if (fabsf(x[i]) > max){
13             iamax = i;
14             max = fabsf(x[i]);
15         }
16     }
17     return iamax;
18 }

```

```

6  static int align_si_amax_avx(int n, float *x, int nprol, int nvecl)
7  {
8      float max = 0.0;
9      int i, j, iamax, ind;
10     float __attribute__((aligned(64))) vindx[8];
11     __m256 v_x, v_neg, ind_ant, var_ant, var_max, ind_new, mask, indand, indnand;
12
13     max = fabsf(x[0]);
14     iamax = 0;
15
16     //Prologo
17     for(i=0; i<nprol; i++){
18         if (fabsf(x[i]) > max){
19             iamax = i;
20             max = fabsf(x[i]);
21         }
22     }
23     ind_ant = _mm256_set1_ps((float)iamax);
24     var_ant = _mm256_set1_ps((float)max);
25     ind_new = _mm256_setr_ps(nprol, nprol+1, nprol+2, nprol+3,
26                             nprol+4, nprol+5, nprol+6, nprol+7);
27     //Cuerpo
28     for (i = nprol; i < nvecl; i+=8){
29         v_x = _mm256_load_ps(&x[i]);
30         v_neg = _mm256_sub_ps(_mm256_setzero_ps(), v_x);
31         v_x = _mm256_max_ps(v_x, v_neg);
32         var_max = _mm256_max_ps(v_x, var_ant);
33
34         mask = _mm256_cmp_ps(var_ant, var_max, 0); //es cmpeq
35
36         indand = _mm256_and_ps(mask, ind_ant);
37         indnand = _mm256_andnot_ps(mask, ind_new);
38
39         ind_ant = _mm256_add_ps(indand, indnand);
40         ind_new = _mm256_add_ps(ind_new, _mm256_set1_ps(8));
41         var_ant = var_max;
42     }
43     _mm256_store_ps(&vindx[0], ind_ant);
44
45     for(i=0; i<8; i++){
46         ind = vindx[i];
47         if(fabsf(x[ind]) > max){
48             max = fabsf(x[ind]);
49             iamax = ind;
50         } else if(fabsf(x[ind]) == max && ind < iamax){
51             max = fabsf(x[ind]);
52             iamax = ind;
53         }
54     }
55
56     //Epilogo
57     for(i = nvecl; i < n ; i++){
58         if (fabsf(x[i]) > max){
59             iamax = i;
60             max = fabsf(x[i]);
61         }
62     }
63     return iamax;
64 }

```


TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	7.081375	11.834115	0.034174	0.046706	0.052068
4	1000	6.397300	11.593388	0.081212	0.071771	0.071918
4	1500	6.458528	11.728387	0.105844	0.104807	0.107289
4	2000	3.847041	10.981037	0.178302	0.140208	0.143406
4	2500	6.559700	12.460444	0.171440	0.178121	0.171059
4	3000	6.575183	12.708999	0.217299	0.222359	0.237841
4	3500	6.589551	12.642260	0.234429	0.246486	0.230693
4	4000	6.137499	12.810499	0.274268	0.218577	0.279415
4	8000	6.620101	13.074553	0.510714	0.532566	0.475134
4	13000	6.670709	13.220511	0.829426	0.842502	0.789203
4	18000	6.649394	13.209569	1.325088	1.070181	1.202628
4	23000	6.586503	13.234721	1.348367	1.316639	1.613291
4	28000	7.090113	14.124890	1.661769	2.058182	1.861853
4	33000	6.743828	12.994072	1.624665	2.047646	2.186546
4	38000	6.660429	13.104169	1.841940	2.255571	2.450160
4	43000	6.663634	13.284947	2.427900	2.084481	2.697106
4	48000	6.959185	13.882163	2.858961	2.594294	3.105950
4	53000	6.683786	12.283802	2.454956	3.074199	3.278536
4	58000	6.728707	12.206485	2.577879	3.254581	3.408992
4	100000	6.684504	13.276874	3.677436	4.630643	5.443948
4	175000	6.686002	13.376378	4.630509	6.076806	7.690325
4	250000	6.710364	13.381554	5.306238	8.671990	10.981802
4	325000	6.497108	13.463110	5.635569	9.718861	13.239728
4	400000	6.688786	13.383114	6.005482	8.531947	14.123189
4	475000	6.566406	12.294221	9.459562	13.198176	15.995381
4	550000	6.706155	13.372842	6.508956	14.403799	17.686424
4	625000	6.685566	13.012764	8.210668	11.841250	17.400255
4	700000	6.714271	13.459320	10.146186	15.543887	20.582698
4	775000	6.410004	13.203840	8.088749	13.768194	21.506087
4	1600000	6.692307	12.907123	12.190114	14.657662	27.084823



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	2,885057	6,182266	0,044149	0,038966	0,027405
	2	1000	2,833057	6,072684	0,033799	0,03759	0,08159
	2	1500	3,306005	6,484373	0,140552	0,130676	0,166179
	2	2000	2,764875	5,688694	0,150057	0,140509	0,163026
	2	2500	2,86984	6,677714	0,165061	0,233972	0,245004
	2	3000	2,910949	6,73767	0,222252	0,204586	0,238877
	2	3500	2,827249	5,478626	0,154651	0,355331	0,366616
	2	4000	2,587883	5,562261	0,32122	0,262303	0,297939
L2	2	8000	3,487583	6,820197	0,572631	0,535236	0,262161
	2	13000	2,709259	5,359667	0,542159	1,004474	0,903372
	2	18000	2,853752	5,227945	0,202982	1,040392	1,258683
	2	23000	2,642731	6,625535	0,710458	1,238203	1,308491
	2	28000	2,77037	5,596384	0,842048	1,630428	1,847251
L3	2	33000	2,992613	6,634754	0,851536	1,337236	1,830426
	2	38000	3,331832	6,202233	0,929467	1,407418	1,568754
	2	43000	3,325023	6,57267	1,069212	1,500927	1,893952
	2	48000	2,965956	6,725934	1,06591	1,677177	1,893218
	2	53000	3,34146	6,651935	1,183551	1,782246	2,063028
	2	58000	3,390217	6,165221	0,834695	1,895606	3,236635
	2	100000	3,339023	6,660564	1,393187	2,5986	3,69937
	2	175000	3,331874	6,655056	1,583697	3,625157	4,933684
	2	250000	3,338623	6,667243	1,689589	4,093172	6,018759
	2	325000	3,290604	6,556827	1,70331	4,408972	6,555787
RAM	2	400000	3,315433	6,587762	1,71075	3,687255	8,227729
	2	475000	3,317729	6,334573	1,670961	4,714426	7,12532
	2	550000	3,290866	6,474148	1,821495	5,419524	9,008494
	2	625000	3,323823	6,43558	1,828797	4,770082	6,727134
	2	700000	3,198458	6,480892	1,855044	5,145958	8,776586
	2	775000	3,322136	6,322499	1,732297	5,609433	9,406316



	th	n	sse	paralelo	par-sse
L1	4	500	1,836629	0,007412	0,007956
	4	1000	1,876401	0,016932	0,018938
	4	1500	1,889589	0,027506	0,029121
	4	2000	1,899837	0,029643	0,030834
	4	2500	1,90564	0,045563	0,045704
	4	3000	1,910657	0,05245	0,054515
	4	3500	1,91352	0,049448	0,050525
	4	4000	1,909693	0,059434	0,061514
	4	8000	1,928411	0,14664	0,144185
L2	4	13000	1,914026	0,271665	0,290252
	4	18000	1,913129	0,27213	0,286835
	4	23000	1,914623	0,472221	0,425144
	4	28000	1,918317	0,442736	0,502768
	4	33000	1,925185	0,575707	0,753902
	4	38000	1,92758	0,560267	0,620265
	4	43000	1,919356	0,7553	0,792158
	4	48000	1,921816	0,886553	0,955509
	4	53000	1,915448	0,682756	0,838286
	4	58000	1,916758	1,00647	1,157884
	4	100000	1,9088	1,092005	1,479834
	4	175000	1,920253	1,085393	1,731519
	4	250000	1,840308	1,253464	2,031606
	4	325000	2,488401	1,801294	3,339679
	4	400000	2,231294	1,609498	3,034711
	4	475000	2,049685	1,661251	3,197915
	4	550000	1,913079	1,551517	2,820149
	4	625000	1,71407	1,562341	2,857535
	4	700000	1,693736	1,417645	2,1923
4	775000	1,690043	1,540933	2,712045	



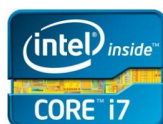
di-amax

```

4  int di_amax_seq(int n, double *x)
5  {
6      double max;
7      int i, iamax;
8
9      max = fabs(x[0]);
10     iamax = 0;
11     for(i=1; i<n; i++){
12         if (fabs(x[i]) > max) {
13             iamax = i;
14             max = fabs(x[i]);
15         }
16     }
17     return iamax;
18 }

```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	3.232156	6.149807	0.035055	0.035127	0.030258
4	1000	3.279716	6.418375	0.071181	0.063981	0.076835
4	1500	3.280579	6.457192	0.099749	0.107377	0.108030
4	2000	3.307373	6.535771	0.159125	0.146102	0.150738
4	2500	3.309311	6.582306	0.145448	0.185138	0.173260
4	3000	3.311482	6.584571	0.197059	0.193807	0.194214
4	3500	3.316198	6.594394	0.238126	0.224419	0.207821
4	4000	3.317401	6.594522	0.250310	0.294965	0.256280
4	8000	3.365854	6.718954	0.532353	0.523357	0.510659
4	13000	3.328547	6.233342	0.650853	0.723917	0.807076
4	18000	3.330043	6.645372	0.768668	1.151330	1.286690
4	23000	3.330438	6.654242	0.902949	1.373253	1.456466
4	28000	3.316467	6.658501	0.974848	1.370884	1.488645
4	33000	3.304502	6.669709	1.064551	1.637141	2.077665
4	38000	3.334341	6.743271	1.189450	2.272160	1.957257
4	43000	3.366923	6.682237	1.283771	2.218971	2.083302
4	48000	3.301808	6.613713	1.655231	2.411977	2.692808
4	53000	3.339406	6.619375	1.307310	2.588165	2.606117
4	58000	3.346963	6.652476	1.655596	2.677624	3.793209
4	100000	3.196639	6.718295	1.519158	3.120460	4.757497
4	175000	3.241099	7.314510	2.619279	5.164003	6.690694
4	250000	3.318954	6.693475	3.072634	6.704861	7.951108
4	325000	3.281758	6.723256	2.848495	7.328536	9.340541
4	400000	3.309536	6.761950	2.814464	7.856376	10.924214
4	475000	3.287591	6.709224	3.153946	8.299835	12.192297
4	550000	3.350430	6.697673	3.376893	8.222280	12.580191
4	625000	3.111383	6.697532	3.228134	9.396288	13.560880
4	700000	3.331944	6.639524	3.547564	9.084070	13.326186
4	775000	3.238322	6.476219	3.278192	9.103998	13.271594
4	1600000	3.318772	6.198347	3.381062	9.673039	14.690114



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	2,885057	6,182266	0,044149	0,038966	0,027405
	2	1000	2,833057	6,072684	0,033799	0,03759	0,08159
	2	1500	3,306005	6,484373	0,140552	0,130676	0,166179
	2	2000	2,764875	5,688694	0,150057	0,140509	0,163026
	2	2500	2,86984	6,677714	0,165061	0,233972	0,245004
	2	3000	2,910949	6,73767	0,222252	0,204586	0,238877
	2	3500	2,827249	5,478626	0,154651	0,355331	0,366616
	2	4000	2,587883	5,562261	0,32122	0,262303	0,297939
L2	2	8000	3,487583	6,820197	0,572631	0,535236	0,262161
	2	13000	2,709259	5,359667	0,542159	1,004474	0,903372
	2	18000	2,853752	5,227945	0,202982	1,040392	1,258683
	2	23000	2,642731	6,625535	0,710458	1,238203	1,308491
	2	28000	2,77037	5,596384	0,842048	1,630428	1,847251
L3	2	33000	2,992613	6,634754	0,851536	1,337236	1,830426
	2	38000	3,331832	6,202233	0,929467	1,407418	1,568754
	2	43000	3,325023	6,57267	1,069212	1,500927	1,893952
	2	48000	2,965956	6,725934	1,06591	1,677177	1,893218
	2	53000	3,34146	6,651935	1,183551	1,782246	2,063028
	2	58000	3,390217	6,165221	0,834695	1,895606	3,236635
	2	100000	3,339023	6,660564	1,393187	2,5986	3,69937
	2	175000	3,331874	6,655056	1,583697	3,625157	4,933684
	2	250000	3,338623	6,667243	1,689589	4,093172	6,018759
	2	325000	3,290604	6,556827	1,70331	4,408972	6,555787
RAM	2	400000	3,315433	6,587762	1,71075	3,687255	8,227729
	2	475000	3,317729	6,334573	1,670961	4,714426	7,12532
	2	550000	3,290866	6,474148	1,821495	5,419524	9,008494
	2	625000	3,323823	6,43558	1,828797	4,770082	6,727134
	2	700000	3,198458	6,480892	1,855044	5,145958	8,776586
	2	775000	3,322136	6,322499	1,732297	5,609433	9,406316



	th	n	sse	paralelo	par-sse
L1	4	500	1,216667	0,007692	0,007763
	4	1000	1,21338	0,014862	0,015607
	4	1500	1,213889	0,021253	0,022986
	4	2000	1,180698	0,035634	0,035174
	4	2500	1,211472	0,019482	0,035216
	4	3000	1,210875	0,05216	0,054023
	4	3500	1,146921	0,067104	0,075157
	4	4000	1,216268	0,073732	0,071236
L2	4	8000	1,205876	0,151053	0,139926
	4	13000	1,208613	0,20357	0,213817
	4	18000	1,211035	0,259163	0,299031
	4	23000	1,212701	0,364735	0,411606
	4	28000	1,209982	0,423426	0,462757
	4	33000	1,210059	0,445619	0,46008
	4	38000	1,201052	0,50806	0,562394
	4	43000	1,210044	0,479992	0,609006
	4	48000	1,211041	0,788257	0,921397
	4	53000	1,210006	0,618116	0,683639
	4	58000	1,216097	0,605348	0,711041
	4	100000	1,615887	0,958	1,011942
	4	175000	1,190497	1,108467	1,470012
	4	250000	1,360852	1,35884	1,809336
	4	325000	1,09392	1,301881	1,669816
	4	400000	1,17263	1,36932	1,272624
	4	475000	1,188026	1,38847	1,540326
	4	550000	1,116868	1,317675	1,295464
4	625000	1,176523	1,323343	1,136754	
4	700000	1,641875	1,906458	2,549981	
4	775000	1,282177	1,53772	1,516647	



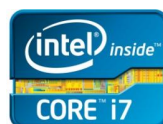
norm2

snorm2

```
4 float snrm2_seq(int n, float *x)
5 {
6     int i;
7     float scale, ssq, absol, norm, w;
8
9     if(n <= 1) norm = 0.0;
10    else if(n == 1) norm = fabsf(x[0]);
11    else{
12        scale = 0.0;
13        ssq = 1.0;
14        for(i=0; i<n; i++){
15            if(x[i] != 0){
16                absol = fabsf(x[i]);
17                if(scale < absol){
18                    w = scale/absol;
19                    ssq = 1.0+ssq*w*w;
20                    scale = absol;
21                }else{
22                    w = absol/scale;
23                    ssq = ssq + w*w;
24                }
25            }
26        }
27        norm = scale*(float)sqrt((double)ssq);
28    }
29
30    return norm;
31 }
32
33 float snrm2_seq2(int n, float *x)
34 {
35     int ind, i;
36     float norm = 0.0, max = 0.0;
37     float sum = 0.0, mult;
38
39     max = fabsf(x[0]);
40     for(i=1; i<n; i++)
41         if(fabsf(x[i]) > max)
42             max = fabsf(x[i]);
43
44     mult = 1.0/max;
45
46     for(i=0; i<n; i++)
47         sum = sum + (x[i]*mult) * (x[i]*mult);
48
49     norm = (float)sqrt((double)sum);
50
51     return max*norm;
52 }
```

```
10 static float th_omp_max(float *x, int desde, int hasta)
11 {
12     int i;
13     float max = 0;
14     for (i=desde; i<hasta; i++)
15     {
16         if (fabsf(x[i])> max)
17             max = fabsf(x[i]);
18     }
19     return max;
20 }
21 float snrm2_omp(int n, float *x)
22 {
23     int i, ini, fin;
24     float max = 0.0;
25     float norm = 0.0;
26     float sum = 0.0;
27     float mult;
28
29     int trozo=ceil((float)n/(float)NTHR);
30     float *w;
31     w=malloc(NTHR*sizeof(float));
32
33     #pragma omp parallel private (i,ini,fin) firstprivate (n,trozo)
34     {
35         i = omp_get_thread_num();
36         ini=i*trozo;
37         fin=min(trozo*(i+1),n);
38         w[i] = th_omp_max(x,ini,fin);
39     }
40     for (i=0; i<NTHR; i++)
41         if (max<w[i]) max=w[i];
42     free(w);
43
44     mult = 1.0/max;
45
46     #pragma omp parallel for reduction(+:sum) firstprivate(mult)
47     for(i=0;i<n;i++)
48         sum = sum + (x[i]*mult)*(x[i]*mult);
49
50     norm = (float)sqrt((double)sum);
51
52     return max*norm;
53 }
```


TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	10.130533	15.827360	13.113342	0.027774	0.032884	0.031557	0.037209
4	1000	11.063579	19.698652	15.564668	0.060746	0.082836	0.058708	0.064212
4	1500	11.370598	20.368330	15.994063	0.096668	0.086919	0.095858	0.082301
4	2000	11.526846	21.727807	16.744842	0.146803	0.145606	0.113470	0.128664
4	2500	11.636021	21.692594	16.708196	0.158066	0.149665	0.181072	0.179972
4	3000	11.673304	22.435108	17.111041	0.180965	0.169926	0.213728	0.275518
4	3500	11.667745	22.018601	17.458596	0.239744	0.235952	0.254348	0.284244
4	4000	13.445612	26.068519	19.809134	0.284431	0.266749	0.336928	0.390010
4	8000	11.862716	23.377581	15.357280	0.402728	0.470875	0.552556	0.592525
4	13000	12.325697	23.756605	18.072662	0.648188	0.861169	1.001239	0.882303
4	18000	12.118679	23.537398	17.886725	0.740047	1.185774	1.347745	1.242054
4	23000	11.870984	23.504405	17.652883	1.088107	1.369741	1.384338	1.456791
4	28000	11.976956	23.582446	17.843856	0.604474	1.761264	1.861816	1.772804
4	33000	12.145517	23.854252	16.257901	1.376192	2.163606	2.314590	2.012111
4	38000	12.125021	23.856310	18.067856	1.306333	1.970483	2.661181	2.893386
4	43000	12.115230	16.871772	15.696665	1.464834	2.466257	2.610024	2.592221
4	48000	12.122611	23.607756	18.082284	1.564552	2.855853	2.786109	2.943273
4	53000	12.032040	23.746116	17.947797	1.600380	3.098888	3.067377	3.083410
4	58000	12.144402	23.997745	17.881495	1.906785	3.087964	3.083291	3.221275
4	100000	12.132943	21.121067	16.297280	1.829658	5.187420	5.266719	5.808124
4	175000	12.126347	21.827335	17.303604	2.403193	8.242289	8.793370	7.957256
4	250000	11.655686	22.190777	17.461146	3.220825	9.286700	11.595089	12.457296
4	325000	12.008680	22.252755	16.996015	3.716525	11.142802	14.114085	12.539003
4	400000	12.082541	22.571442	17.426343	3.714288	13.140233	15.079183	14.425455
4	475000	12.001812	22.583181	17.536852	4.439506	14.358412	17.553346	16.529946
4	550000	12.038657	22.519296	17.687800	3.939306	16.825384	19.063655	18.104533
4	625000	12.182270	23.022992	17.910926	2.745606	11.781537	21.207437	20.790571
4	700000	11.935608	22.624019	17.583955	4.217118	19.074757	22.995399	21.516035
4	775000	12.026458	22.636539	17.706463	4.287113	20.679167	25.102006	22.848476
4	1600000	11.392987	19.761748	16.492177	4.459404	23.302831	27.389535	32.599311



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma	serie2
L1	2	500	10,31464	15,857143	13,072314	0,032605	0,047863	0,047733	0,047707	0,073094
	2	1000	11,21578	19,663351	15,646055	0,057319	0,065245	0,082149	0,102871	0,078943
	2	1500	11,459547	20,624702	16,002522	0,085196	0,098507	0,096091	0,098321	0,079468
	2	2000	9,03481	17,495158	16,096366	0,151574	0,214297	0,203995	0,214218	0,086126
	2	2500	11,087799	19,166886	16,831618	0,209892	0,194081	0,212738	0,278311	0,087776
	2	3000	9,861546	19,450212	16,032298	0,188768	0,285207	0,293487	0,295064	0,075772
	2	3500	9,417601	17,745053	16,317485	0,274108	0,308598	0,340273	0,364008	0,085573
	2	4000	9,902161	18,843191	17,808896	0,380625	0,363205	0,431896	0,426646	0,088495
	2	8000	11,933892	23,446411	17,739964	0,401801	0,687361	0,668841	0,659623	0,079542
L2	2	13000	12,133345	23,392294	17,545451	0,737148	1,178909	0,882846	1,051425	0,080533
	2	18000	12,124772	23,589647	17,905143	0,882893	1,282625	1,551041	1,529932	0,08034
	2	23000	12,340654	24,11396	18,271104	0,99773	1,738478	2,170081	2,176377	0,081465
	2	28000	12,081319	16,342889	15,372177	0,981867	2,463861	2,21524	2,145863	0,077983
	2	33000	10,058491	18,335702	14,069577	1,286067	2,163283	2,611952	3,117068	0,076226
	2	38000	9,238672	16,860799	15,879718	1,299797	2,355913	3,278502	3,258823	0,073265
	2	43000	12,126438	23,901448	18,065628	1,249913	3,003374	3,663576	2,933619	0,079888
	2	48000	11,840011	24,000172	18,133123	1,332229	3,170259	4,098832	3,995257	0,080222
	2	53000	12,131991	21,720635	18,100141	1,383526	3,31663	4,388215	4,333212	0,080052
	2	58000	9,05716	16,067392	15,319436	1,387016	3,285948	3,606198	4,092337	0,081217
L3	2	100000	12,097456	21,565129	16,490495	1,715961	4,658036	5,985838	5,495531	0,075595
	2	175000	9,762393	16,982465	15,695327	2,1605	7,475276	8,731207	6,963111	0,078401
	2	250000	12,210076	23,564446	18,240705	2,457527	7,576122	11,090757	9,032424	0,079452
	2	325000	9,66035	22,529294	17,550372	2,357209	8,826086	11,485858	10,818669	0,077343
	2	400000	12,241735	23,150813	17,427708	2,43223	9,988158	12,318142	11,644399	0,078559
	2	475000	11,99635	23,314644	18,036186	2,518287	12,373151	14,466077	12,472633	0,078492
	2	550000	12,047031	22,162764	17,701666	2,445546	11,533943	14,174168	14,829858	0,077292
	2	625000	12,003615	22,384505	17,772842	2,55335	13,674935	16,198973	15,599243	0,078187
	2	700000	11,433659	22,028137	17,573788	2,54768	14,959604	18,072726	15,467456	0,07848
	2	775000	11,549549	20,436337	17,083047	2,54558	12,68028	15,559916	15,798439	0,078535



	th	n	sse	paralelo	par-sse	serie nuevo
L1	4	500	11,532563	0,019844	0,019731	0,156574
	4	1000	12,756678	0,042122	0,049135	0,162316
	4	1500	12,922745	0,065071	0,062431	0,163435
	4	2000	13,311312	0,082151	0,086052	0,164378
	4	2500	13,615956	0,104203	0,104073	0,163807
	4	3000	13,75323	0,157384	0,154982	0,164937
	4	3500	13,795595	0,180615	0,17316	0,164536
	4	4000	13,850685	0,1743	0,174238	0,164578
	4	8000	13,865866	0,403378	0,421036	0,162291
L2	4	13000	12,437709	0,665701	0,670904	0,198244
	4	18000	12,430542	0,788757	0,728286	0,165486
	4	23000	12,432833	0,878818	1,038413	0,165066
	4	28000	15,458166	1,468995	1,467075	0,210212
	4	33000	16,607264	1,809106	2,155332	0,220613
	4	38000	12,444636	1,618729	1,966001	0,165509
	4	43000	16,848791	2,407973	2,879042	0,234288
	4	48000	12,502077	1,766075	2,07449	0,165951
	4	53000	17,681106	2,817462	3,759965	0,234702
	4	58000	15,76859	3,003253	3,944603	0,234236
	4	100000	12,52816	2,579128	3,654022	0,166784
	4	175000	12,281515	3,483651	6,739413	0,166445
	4	250000	12,415792	3,770925	7,369161	0,165784
	4	325000	12,356621	3,986683	8,350328	0,165727
	4	400000	12,385828	4,111144	8,860171	0,164788
	4	475000	12,430597	4,224015	10,904203	0,164278
	4	550000	11,510212	4,23809	10,401137	0,165739
	4	625000	12,36559	4,337575	10,406182	0,164568
4	700000	12,335011	4,409674	8,434678	0,165321	
4	775000	11,845248	4,457866	9,096816	0,165728	



dnorm2

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	5.640417	10.326531	8.071006	0.032939	0.033297	0.034918	0.033001
4	1000	6.151496	11.806395	9.058824	0.062615	0.049757	0.063640	0.075096
4	1500	5.851532	11.383826	8.660698	0.090362	0.090199	0.107312	0.137636
4	2000	5.892765	11.529088	8.746785	0.120703	0.114401	0.131822	0.177827
4	2500	5.910567	11.622848	8.800537	0.148197	0.145378	0.190938	0.217956
4	3000	5.696044	11.691584	8.830348	0.174696	0.171612	0.217489	0.262443
4	3500	5.938799	11.715007	8.698283	0.206000	0.197372	0.210283	0.208597
4	4000	5.960732	11.760767	8.900455	0.237650	0.252536	0.231108	0.241741
4	8000	6.263041	12.128936	9.602929	0.456809	0.484124	0.623901	0.651944
4	13000	5.878101	11.698094	8.872048	0.643573	0.873470	0.941486	0.961368
4	18000	5.997440	11.416376	8.921831	0.826642	1.051083	1.020581	1.091787
4	23000	6.049972	9.845879	9.021377	0.999254	1.559341	1.618856	1.666298
4	28000	5.901680	11.839199	8.940881	1.157597	1.597451	1.770881	1.942667
4	33000	6.045603	11.731223	8.970597	1.256529	1.679768	1.919477	1.973276
4	38000	6.070342	10.498280	8.469538	1.341998	2.180850	1.906546	1.866656
4	43000	5.549251	9.842565	8.146249	1.416639	2.301500	2.243158	2.463171
4	48000	6.079260	10.558767	8.325988	1.600576	2.350453	2.627377	2.608663
4	53000	6.098739	10.065079	8.453152	1.554736	2.848339	2.816167	2.823600
4	58000	5.946310	10.618883	8.395171	1.637343	3.013505	2.862648	2.982317
4	100000	5.775519	10.947035	8.540671	1.927024	4.533812	4.970990	4.002800
4	175000	5.986191	10.694970	8.391803	2.909965	6.004374	6.407900	6.371797
4	250000	5.977755	10.950288	8.817446	3.419936	7.634343	8.710946	8.458209
4	325000	5.778296	11.389923	8.873662	3.840273	8.976467	10.605401	10.040015
4	400000	6.082285	11.314655	8.949822	4.077615	10.316800	12.183895	11.956565
4	475000	5.949333	10.969685	8.705205	3.883393	10.891921	13.938432	12.623652
4	550000	5.941544	10.941669	8.764740	3.781455	10.923932	13.755523	10.530823
4	625000	5.937748	10.516532	8.510615	4.509553	12.250691	15.084589	10.910098
4	700000	5.934785	10.365637	8.497220	4.218761	10.718307	10.491022	12.015687
4	775000	5.702506	9.763448	8.293865	4.245031	8.345539	14.390396	15.787847
4	1600000	5.358445	7.880629	7.603458	4.805455	13.282841	16.228225	15.964023



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma	serie2
L1	2	500	4,91711	9,222402	8,131604	0,033784	0,034777	0,042607	0,054944	0,084374
	2	1000	5,783998	11,09306	8,513349	0,060545	0,090179	0,088318	0,089119	0,078391
	2	1500	5,872285	11,372951	8,607006	0,085982	0,094267	0,094648	0,094508	0,078691
	2	2000	5,889087	11,507181	8,741747	0,11306	0,124242	0,125382	0,125196	0,077578
	2	2500	5,705891	11,630607	8,800503	0,139544	0,15539	0,156284	0,17921	0,078475
	2	3000	5,926557	11,684351	8,829123	0,169068	0,185205	0,185058	0,185702	0,078821
	2	3500	4,962042	9,633709	8,755294	0,216028	0,279715	0,372063	0,389971	0,089686
	2	4000	6,312392	12,488277	9,415904	0,229298	0,261891	0,271579	0,336345	0,082887
L2	2	8000	5,96859	11,579106	8,796804	0,391851	0,615322	0,687864	0,659791	0,07914
	2	13000	5,977589	11,714114	7,552481	0,769197	1,067765	1,136648	1,120716	0,078972
	2	18000	5,43407	11,770856	8,907772	0,718052	1,004932	1,120053	1,01821	0,079153
	2	23000	5,990081	11,78914	8,928549	0,860061	1,263035	1,316867	1,314472	0,07899
	2	28000	6,039146	11,91164	9,010379	1,176009	1,739765	1,877998	2,151865	0,073544
L3	2	33000	4,909438	8,365127	6,921675	1,204377	2,107986	2,568744	2,615713	0,082313
	2	38000	4,890572	8,457317	7,773462	1,302194	2,503275	2,898341	2,647618	0,050051
	2	43000	5,760062	8,2634	7,577063	1,20636	2,362745	2,070562	2,351799	0,07942
	2	48000	6,000328	10,590031	8,326651	1,197091	2,233554	2,586971	3,368178	0,079317
	2	53000	6,012369	10,709558	8,402771	1,239267	2,317188	3,193936	3,491419	0,079529
	2	58000	5,987069	10,718171	8,411431	1,32136	2,039162	3,053775	3,031089	0,077194
	2	100000	6,074034	10,94552	8,750016	1,742873	3,135886	3,851411	3,850212	0,052668
	2	175000	5,425163	10,562894	8,446817	2,025526	3,926962	4,854243	5,163605	0,078876
	2	250000	2,018954	10,984069	8,882054	2,19236	5,561289	7,058728	7,087426	0,081686
	2	325000	5,697404	7,769305	8,621063	2,398172	7,084123	9,673696	8,887001	0,076198
RAM	2	400000	5,936533	9,790835	7,57767	2,367616	7,699728	9,918129	9,00262	0,076634
	2	475000	5,874919	10,102905	8,353859	2,370841	7,038172	10,072858	8,394667	0,077775
	2	550000	5,829209	9,185664	8,202739	2,44068	7,163907	10,405327	10,015371	0,077432
	2	625000	5,704247	9,208024	7,977417	2,445895	8,569506	8,882044	8,105124	0,078638
	2	700000	5,792438	9,019087	8,030857	2,4476	7,340935	7,885767	5,130309	0,079428
	2	775000	5,774622	8,936303	7,60815	2,529846	7,486191	4,783517	8,845555	0,079094



	th	n	sse	paralelo	par-sse	serie nuevo
L1	4	500	6,481281	0,02429	0,023228	0,159222
	4	1000	6,742945	0,040065	0,039707	0,161879
	4	1500	6,861046	0,071665	0,070272	0,163056
	4	2000	6,926173	0,09308	0,094396	0,163192
	4	2500	6,941528	0,129987	0,141373	0,23082
	4	3000	6,982005	0,142607	0,139016	0,163249
	4	3500	6,985647	0,177103	0,178779	0,170578
	4	4000	6,898337	0,190693	0,193496	0,163506
L2	4	8000	6,151393	0,328673	0,321233	0,163711
	4	13000	6,184379	0,642848	0,641572	0,165048
	4	18000	6,173616	0,765578	0,715028	0,164705
	4	23000	6,191183	1,052517	1,201075	0,233478
	4	28000	6,192101	1,425744	1,700568	0,233183
	4	33000	8,100842	1,626943	1,787448	0,215695
	4	38000	6,190046	1,683753	2,502228	0,233409
	4	43000	6,173472	1,802155	2,362552	0,233255
	4	48000	6,215985	1,706305	1,841463	0,164633
	4	53000	8,30012	2,476873	2,787085	0,221076
	4	58000	6,173676	1,958123	2,485201	0,164697
	4	100000	6,187168	2,480852	3,018183	0,16488
	4	175000	5,847044	2,93217	4,302701	0,165243
	4	250000	6,156114	3,181203	4,144359	0,164591
	4	325000	6,122535	3,394312	5,801472	0,165235
	4	400000	6,15106	3,274323	4,385334	0,164651
	4	475000	6,166299	3,095552	5,356541	0,165807
	4	550000	5,853423	3,935868	4,831191	0,162702
4	625000	5,308315	3,909361	4,481014	0,165521	
4	700000	4,876991	3,96411	5,05185	0,164495	
4	775000	4,911938	3,469624	4,316774	0,166042	



asum

iasum

```

4  int iasum_seq(int n, int *x)
5  {
6      int temp = 0;
7      int i;
8
9      for(i=0;i<n;i++){
10         temp = temp + abs(x[i]);
11     }
12     return temp;
13 }

```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	4.915946	8.443737	0.006102	0.008112	0.007216
4	1000	5.641912	9.345920	0.015606	0.016691	0.019413
4	1500	6.814844	12.957511	0.029548	0.022131	0.040232
4	2000	5.748770	10.677442	0.032408	0.036828	0.036668
4	2500	5.873882	10.966609	0.049590	0.048994	0.038610
4	3000	5.571219	10.803987	0.043053	0.056520	0.056626
4	3500	5.914286	11.313407	0.055460	0.016745	0.048937
4	4000	5.843429	10.829627	0.057971	0.026489	0.051402
4	8000	5.964219	11.336128	0.109533	0.136622	0.128050
4	13000	5.367940	6.401477	0.197698	0.191786	0.193505
4	18000	5.807724	8.044747	0.207389	0.280453	0.290511
4	23000	5.510417	7.878838	0.351495	0.279333	0.345353
4	28000	6.212996	9.230176	0.401836	0.413134	0.441958
4	33000	6.222373	9.218732	0.465746	0.471499	0.454448
4	38000	5.201945	8.732881	0.464155	0.515434	0.509071
4	43000	6.279191	9.746417	0.582136	0.641830	0.698884
4	48000	6.530390	10.856594	0.714194	0.710474	0.771777
4	53000	5.584094	8.564440	0.687070	0.802504	0.789339
4	58000	5.780553	9.644971	0.716801	0.758962	0.669321
4	100000	4.727817	5.934087	0.878556	1.319868	1.440341
4	175000	5.448359	6.621513	1.568893	2.360439	2.293030
4	250000	4.878576	5.986824	1.557365	3.080122	3.008432
4	325000	5.026647	6.071518	1.478213	3.497813	3.871850
4	400000	4.931068	6.251364	1.411950	3.792004	3.456234
4	475000	5.175195	6.054390	1.540716	4.399198	4.961002
4	550000	4.721282	6.188087	2.244839	5.366321	5.364587
4	625000	5.325289	6.460147	2.400508	5.796827	5.581739
4	700000	5.025970	6.081645	2.289285	4.450102	6.452627
4	775000	5.077177	6.375518	1.968079	6.164045	6.451980
4	1600000	4.487717	5.973639	2.106507	6.646305	8.873554



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	5,120466	8,117043	0,006961	0,007976	0,007733
	2	1000	5,100697	8,114424	0,018527	0,019199	0,024487
	2	1500	5,87354	6,735585	0,025237	0,018768	0,022152
	2	2000	4,006696	7,169585	0,028645	0,028572	0,031312
	2	2500	5,928129	10,750562	0,047813	0,045871	0,058856
	2	3000	5,914971	11,441978	0,04096	0,041637	0,044753
	2	3500	6,136928	14,576923	0,096049	0,08198	0,090705
	2	4000	3,625149	11,197713	0,056819	0,083723	0,056368
	2	8000	5,366205	10,413075	0,275751	0,308482	0,167263
L2	2	13000	5,377836	6,380418	0,226637	0,283092	0,288199
	2	18000	5,467541	7,270682	0,228434	0,226463	0,274847
	2	23000	3,439414	6,350922	0,295721	0,452585	0,393425
	2	28000	6,145498	11,419756	0,742731	1,173807	1,129087
	2	33000	5,602293	7,692152	0,313853	0,530278	0,550274
	2	38000	3,651924	6,261264	0,466137	0,652428	0,660754
	2	43000	4,325478	8,627383	0,508983	0,849662	0,892413
	2	48000	5,344566	9,048763	0,850372	1,102567	1,409488
	2	53000	3,394827	5,017845	0,496729	0,997676	1,063841
L3	2	58000	2,92513	2,960151	0,449592	1,038654	1,122946
	2	100000	5,650757	6,710654	0,809254	1,90308	1,654736
	2	175000	5,378613	8,505238	1,413234	3,580122	3,772102
	2	250000	5,455341	8,413724	1,435339	4,393953	5,224546
	2	325000	4,332689	6,002472	1,029353	2,566014	2,581704
	2	400000	8,326956	11,533715	2,268344	5,222454	5,578356
	2	475000	9,114913	10,383353	1,760256	5,362133	6,651941
	2	550000	5,170053	5,989411	0,994659	3,652667	3,787279
	2	625000	5,210785	6,469465	1,354462	3,806535	4,207494
2	700000	5,493798	6,79658	1,353711	5,046172	5,258175	
2	775000	6,626612	7,771617	1,976836	6,063766	6,83476	



	th	n	sse	paralelo	par-sse
L1	4	500	1,963351	0,00451	0,007176
	4	1000	2,357085	0,009429	0,009844
	4	1500	2,568067	0,021008	0,022314
	4	2000	2,672085	0,017437	0,019091
	4	2500	2,744438	0,021293	0,020634
	4	3000	2,772082	0,02385	0,025531
	4	3500	2,815827	0,032549	0,033586
	4	4000	2,844688	0,029229	0,035612
	4	8000	2,897834	0,06186	0,059666
L2	4	13000	2,486997	0,100071	0,095549
	4	18000	2,507837	0,145557	0,134335
	4	23000	2,527962	0,181337	0,193445
	4	28000	2,527734	0,198765	0,225106
	4	33000	2,52303	0,337265	0,37874
	4	38000	2,522773	0,277694	0,290027
	4	43000	2,544686	0,36407	0,439327
	4	48000	2,555324	0,352198	0,381641
	4	53000	2,55427	0,438963	0,459395
	4	58000	2,548232	0,37989	0,469522
	4	100000	2,612468	0,739172	0,914931
	4	175000	3,242492	1,106157	1,336175
	4	250000	2,923921	1,009675	1,323714
	4	325000	2,55324	0,816696	1,266589
	4	400000	1,906337	1,096809	1,321833
	4	475000	2,652	1,033995	1,534835
	4	550000	2,657761	1,140105	1,752792
4	625000	1,90916	1,225554	1,576875	
4	700000	1,851219	1,201009	1,544038	
4	775000	1,957861	1,215632	1,676861	



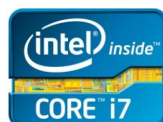
sasum

```

4 float sasum_seq(int n, float *x)
5 {
6     float temp = 0.0;
7     int i;
8
9     for(i=0;i<n;i++){
10        temp = temp + fabsf(x[i]);
11    }
12    return temp;
13 }

```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	3.755602	7.056579	0.009096	0.012343	0.012816
4	1000	3.896729	7.461647	0.021436	0.020980	0.020767
4	1500	3.916976	7.595592	0.031038	0.031503	0.032553
4	2000	4.136918	8.483646	0.044332	0.046788	0.049170
4	2500	3.913734	7.748089	0.053087	0.048445	0.056902
4	3000	3.965249	7.817710	0.060765	0.065268	0.079763
4	3500	3.965261	7.861194	0.079391	0.074859	0.067761
4	4000	3.974606	7.824382	0.080439	0.083763	0.082848
4	8000	3.988813	7.912733	0.143983	0.174466	0.166600
4	13000	3.984980	7.591788	0.245359	0.253071	0.267229
4	18000	4.116374	7.955150	0.359060	0.371000	0.353085
4	23000	3.991938	7.749455	0.356711	0.446517	0.419361
4	28000	3.989341	7.794235	0.428127	0.570741	0.593943
4	33000	3.988862	7.824058	0.458241	0.534981	0.634821
4	38000	4.013761	7.892354	0.523385	0.717776	0.799232
4	43000	4.018853	7.932750	0.571564	0.905489	0.813570
4	48000	3.263050	7.892127	0.713562	0.820963	0.945303
4	53000	4.227758	8.349568	0.839050	0.947290	1.055483
4	58000	3.638163	7.800860	0.743441	0.931279	1.089243
4	100000	4.017308	6.628374	1.042456	1.632484	1.991372
4	175000	3.804675	6.141582	1.300106	2.786607	2.565239
4	250000	3.993708	7.198189	1.475055	3.427371	4.129783
4	325000	3.974354	7.302824	1.540007	3.181720	4.992826
4	400000	3.993125	7.438208	2.007767	4.268265	5.550488
4	475000	3.934113	7.175837	2.225995	4.700428	7.153975
4	550000	4.180381	7.870000	2.134066	4.824365	6.657624
4	625000	3.995093	7.365090	1.940175	4.990853	6.812897
4	700000	3.990064	7.396287	1.905379	5.045722	6.921881
4	775000	4.036404	7.722606	1.794016	5.057250	7.279073
4	1600000	3.846017	6.848869	2.929527	8.618821	11.699637



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	3,706777	6,836735	0,009304	0,01679	0,016807
	2	1000	3,904674	7,296424	0,017602	0,027189	0,025908
	2	1500	3,259887	6,49096	0,029302	0,050131	0,05037
	2	2000	3,535527	7,148295	0,038549	0,0443	0,066799
	2	2500	3,949716	6,505299	0,041084	0,07359	0,081608
	2	3000	4,002759	7,881975	0,052317	0,098212	0,098975
	2	3500	3,032122	7,045522	0,108127	0,114753	0,115437
	2	4000	3,818272	7,760213	0,069354	0,090313	0,068253
	2	8000	3,866139	7,472597	0,182161	0,250148	0,203797
L2	2	13000	3,666805	7,22325	0,345093	0,384649	0,217091
	2	18000	3,359562	6,214926	0,376224	0,516254	0,47251
	2	23000	3,5664	7,634887	0,419856	0,637941	0,517845
	2	28000	3,987406	7,783408	0,586275	0,611023	0,657429
	2	33000	4,046647	8,038316	0,611326	0,777925	0,629188
	2	38000	3,051678	6,101774	0,64019	0,886177	1,044995
	2	43000	3,428244	6,357258	0,746556	1,057007	1,157818
	2	48000	3,534883	7,603793	0,790656	1,152229	0,963376
	2	53000	3,846567	6,290344	0,644047	1,180896	1,382472
L3	2	58000	3,651007	6,296062	0,991391	1,301517	1,477863
	2	100000	3,947727	6,627847	1,097287	1,510287	1,596921
	2	175000	4,012306	6,700862	1,342904	1,999894	2,798355
	2	250000	3,200025	6,911573	1,369031	2,582787	3,09004
	2	325000	3,883455	8,107969	1,499529	3,162902	3,759868
	2	400000	4,041547	7,582413	1,493079	3,204768	4,086879
	2	475000	4,113315	7,773218	1,575141	3,62913	4,573707
	2	550000	4,020651	7,613247	1,595257	3,833397	4,991854
	2	625000	4,215205	7,990606	1,697176	5,032887	6,75395
2	700000	3,292957	6,043524	1,645199	4,626686	5,529011	
2	775000	3,331286	6,910528	1,738549	4,486216	6,445579	



	th	n	sse	paralelo	par-sse
L1	4	500	3,42097	0,007203	0,007318
	4	1000	3,673913	0,011246	0,010402
	4	1500	3,783657	0,021274	0,021039
	4	2000	3,830349	0,027006	0,029589
	4	2500	3,861751	0,031504	0,028894
	4	3000	3,881116	0,044705	0,044182
	4	3500	3,900851	0,03898	0,040157
	4	4000	3,915772	0,042414	0,047293
	4	8000	3,955742	0,08625	0,086986
L2	4	13000	3,370418	0,206769	0,204459
	4	18000	3,354613	0,21033	0,199431
	4	23000	3,367438	0,2654	0,308297
	4	28000	3,37101	0,278263	0,312586
	4	33000	3,386485	0,372695	0,354275
	4	38000	3,425203	0,414992	0,452071
	4	43000	3,353233	0,550988	0,596227
	4	48000	3,396233	0,491954	0,593409
	4	53000	4,562521	0,711179	0,875508
	4	58000	3,424219	0,57672	0,678633
	4	100000	3,399345	0,887093	1,053048
	4	175000	3,420233	1,149184	1,436902
	4	250000	3,834183	1,571234	1,943842
	4	325000	3,834645	1,669351	2,523432
	4	400000	2,784102	1,511878	2,070768
	4	475000	3,239476	1,527916	2,177743
	4	550000	3,368134	1,487034	2,317905
4	625000	3,373087	1,578024	2,453061	
4	700000	2,500811	1,649164	2,534136	
4	775000	3,109739	1,685934	2,590957	



dasum

```

4  double dasum_seq(int n, double *x)
5  {
6      double temp = 0.0;
7      int i;
8
9      for(i=0;i<n;i++){
10         temp = temp + fabs(x[i]);
11     }
12
13     return temp;
14 }

```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	1.965227	3.692572	0.012381	0.012617	0.011772
4	1000	1.889883	3.935507	0.026988	0.027973	0.030118
4	1500	1.988207	3.903448	0.031311	0.031285	0.031787
4	2000	1.991129	3.936929	0.009996	0.036870	0.048276
4	2500	1.987857	3.946249	0.061267	0.065581	0.062355
4	3000	1.990244	3.967503	0.058833	0.061341	0.063030
4	3500	1.992864	3.954940	0.071512	0.075300	0.073894
4	4000	1.993468	3.972458	0.080046	0.084486	0.083143
4	8000	1.995236	3.831210	0.154713	0.166561	0.180837
4	13000	1.995052	3.708901	0.281638	0.247444	0.324399
4	18000	1.997550	3.921017	0.305401	0.312318	0.304539
4	23000	1.998503	3.937040	0.410152	0.464224	0.434817
4	28000	1.853593	3.858655	0.411443	0.528687	0.473612
4	33000	1.947158	3.824870	0.591334	0.553429	0.542677
4	38000	1.986104	3.379890	0.594745	0.594875	0.698411
4	43000	2.004434	3.248106	0.570441	0.696504	0.824815
4	48000	1.986094	3.192626	0.742776	0.729469	0.866814
4	53000	1.897431	3.188891	0.770311	0.833073	0.937331
4	58000	1.734925	3.294249	0.833002	0.984728	1.028638
4	100000	1.916770	3.524866	0.990802	1.401266	1.586795
4	175000	1.937069	3.675294	1.303043	1.966115	2.821064
4	250000	2.026549	3.837723	2.150983	2.040489	2.812621
4	325000	1.995574	3.713022	2.277951	3.126455	4.126724
4	400000	1.995074	3.752347	2.231865	2.886798	4.874601
4	475000	2.059404	3.896185	2.668183	2.967272	4.302822
4	625000	1.850713	3.458782	2.642501	4.312144	5.752732
4	775000	1.982970	3.766210	2.968513	4.992441	6.249128
4	1600000	1.909178	3.041291	3.324685	4.566268	5.324810



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	1,969485	3,691937	0,011564	0,008391	0,009554
	2	1000	1,975247	3,896109	0,017842	0,021184	0,021205
	2	1500	1,984228	3,897713	0,027574	0,029538	0,031724
	2	2000	1,986817	3,939507	0,035203	0,041433	0,045005
	2	2500	1,993059	3,967858	0,045073	0,055522	0,061396
	2	3000	1,993944	3,960657	0,051506	0,061527	0,068147
	2	3500	1,991196	3,95951	0,090079	0,056586	0,068331
	2	4000	1,993372	3,972262	0,084027	0,076337	0,081897
L2	2	8000	1,995496	3,829315	0,202865	0,235113	0,16629
	2	13000	1,531209	2,931536	0,297171	0,291665	0,250713
	2	18000	1,998711	3,924932	0,375756	0,32974	0,322243
	2	23000	1,999644	3,941239	0,438257	0,53212	0,340946
	2	28000	2,002535	3,958482	0,510773	0,558976	0,414427
L3	2	33000	1,995722	3,856911	0,623465	0,669937	0,500017
	2	38000	1,991533	3,462099	0,680733	0,731124	0,665301
	2	43000	1,537385	2,683463	0,648721	0,828445	0,651399
	2	48000	1,987009	3,27326	0,733514	0,822818	0,789436
	2	53000	1,640773	2,877383	0,684741	0,742109	1,213167
	2	58000	1,614603	2,728231	0,77777	0,930217	0,99721
	2	100000	1,966491	3,595333	1,108918	1,03971	1,566085
	2	175000	1,964492	3,750905	1,215649	1,787956	2,252314
	2	250000	1,884188	3,767595	1,321071	1,827526	2,330161
	2	325000	1,570355	3,57682	1,513451	1,696621	2,606922
RAM	2	400000	1,969668	3,534384	1,435229	2,194285	3,01339
	2	475000	1,95527	3,492289	1,481017	2,294939	3,116586
	2	550000	1,931423	3,347013	1,506287	2,365822	3,196461
	2	625000	1,954516	3,252006	1,660695	2,672647	3,103421
	2	700000	1,929295	3,128521	1,567353	2,500699	3,24222
	2	775000	2,073202	3,34275	1,697343	2,678066	3,472375



	th	n	sse	paralelo	par-sse
L1	4	500	1,550355	0,006773	0,007097
	4	1000	1,72509	0,013659	0,013639
	4	1500	1,835152	0,01675	0,01754
	4	2000	1,874961	0,023047	0,024885
	4	2500	1,898088	0,027565	0,025222
	4	3000	1,857889	0,036986	0,042403
	4	3500	1,875314	0,047472	0,048507
	4	4000	1,841437	0,051253	0,078492
L2	4	8000	1,669539	0,104217	0,090157
	4	13000	1,684776	0,145287	0,140851
	4	18000	1,684982	0,251709	0,237498
	4	23000	1,690535	0,30577	0,325823
	4	28000	1,694463	0,349948	0,420365
	4	33000	1,735113	0,409192	0,503502
	4	38000	1,711167	0,397558	0,418909
	4	43000	1,697727	0,345594	0,392924
	4	48000	1,675532	0,491457	0,485762
	4	53000	1,68986	0,539298	0,540459
	4	58000	1,693138	0,679019	0,81148
	4	100000	2,322726	1,052619	1,133348
	4	175000	1,80664	1,483554	1,403026
	4	250000	2,274579	1,742841	1,485797
	4	325000	1,68454	1,129206	1,172445
	4	400000	1,915393	1,253941	1,48996
	4	475000	1,438833	1,45754	1,365158
	4	550000	0,914816	1,196585	1,122245
4	625000	1,183303	1,221482	1,290572	
4	700000	1,831764	1,351046	1,31244	
	4	775000	1,587059	1,207482	1,374335



swap

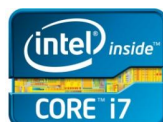
iswap

```

3 void iswap_seq(int n, int *x, int *y)
4 {
5     int i, temp;
6     for (i = 0; i < n; ++i) {
7         temp = x[i];
8         x[i] = y[i];
9         y[i] = temp;
10    }
11 }

```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	3.660000	6.838078	0.006899	0.008815	0.009383
4	1000	4.083871	7.013850	0.014559	0.015006	0.015085
4	1500	4.107875	7.651708	0.022555	0.023411	0.019965
4	2000	4.138420	7.902185	0.029302	0.030044	0.031186
4	2500	4.209530	8.044681	0.036006	0.036136	0.037387
4	3000	4.194249	7.999646	0.041165	0.042360	0.042669
4	3500	4.166824	7.964103	0.043506	0.044857	0.055281
4	4000	4.179997	8.056177	0.049666	0.054561	0.055851
4	8000	2.514727	2.554422	0.123993	0.111669	0.108761
4	13000	2.516013	2.537439	0.178460	0.156535	0.170540
4	18000	2.580781	2.564287	0.162542	0.234478	0.230320
4	23000	2.530939	2.565743	0.280691	0.303060	0.288105
4	28000	2.531603	2.571855	0.369290	0.377087	0.376877
4	33000	2.318461	2.530015	0.443731	0.370937	0.508444
4	38000	2.326408	2.442604	0.511454	0.542631	0.472313
4	43000	2.206246	2.186314	0.552926	0.585465	0.630735
4	48000	2.065071	2.124821	0.608399	0.603206	0.609631
4	53000	2.084598	2.132902	0.571937	0.726026	0.715292
4	58000	2.042874	1.537377	0.698309	0.693529	0.686361
4	100000	2.093686	2.158406	0.894404	1.087779	1.241409
4	175000	2.070705	2.128513	1.379395	1.811265	1.824692
4	250000	2.050620	2.092985	1.246212	1.265393	1.955040
4	325000	2.037929	2.079484	1.537381	2.320679	2.285828
4	400000	2.033444	2.078092	1.216327	2.566843	2.776439
4	475000	1.980440	2.069004	1.566048	2.616062	2.865506
4	550000	1.962072	1.530968	1.125421	3.342266	3.093609
4	625000	1.992793	2.012111	1.714008	2.766668	3.014770
4	700000	1.614639	1.629928	1.931851	3.465991	3.496285
4	775000	1.965802	2.020087	1.879730	2.944836	3.238396
4	1600000	1.966429	2.035974	3.262068	3.096790	3.391762



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	3,549812	5,910798	0,006413	0,011774	0,01184
	2	1000	4,012579	7,440233	0,011826	0,01328	0,017054
	2	1500	2,982811	4,37037	0,019531	0,036184	0,036533
	2	2000	3,763434	7,07266	0,060329	0,062289	0,069866
	2	2500	4,655158	8,943615	0,036845	0,059521	0,065498
	2	3000	3,629019	7,516567	0,046091	0,041481	0,041172
	2	3500	3,979734	7,848016	0,075245	0,079367	0,081436
	2	4000	4,116289	8,001056	0,083779	0,090953	0,09084
L2	2	8000	2,647613	2,546435	0,157277	0,204907	0,16669
	2	13000	2,631147	2,964725	0,248758	0,318616	0,232289
	2	18000	1,933266	2,547369	0,259168	0,284004	0,248922
	2	23000	1,860741	2,329568	0,337434	0,44072	0,271919
	2	28000	1,790171	1,900689	0,307409	0,49304	0,301595
L3	2	33000	2,564068	3,238184	0,566786	0,716963	0,905234
	2	38000	3,077958	3,38781	0,62867	0,720917	0,638697
	2	43000	2,054463	2,479738	0,652717	0,739458	0,625807
	2	48000	1,822016	2,19282	0,562485	0,836738	0,593282
	2	53000	2,738809	3,174935	0,658746	1,116635	1,286808
	2	58000	2,740845	2,483483	0,596053	0,965792	0,870434
	2	100000	2,553623	2,913899	1,13206	1,413066	1,430204
	2	175000	2,415206	2,455416	1,151743	1,59845	1,603925
	2	250000	2,135368	2,216372	1,157155	1,602391	1,549832
RAM	2	325000	3,29638	3,153667	1,961825	3,013076	3,13618
	2	400000	1,901227	1,897632	1,297589	1,828698	1,962032
	2	475000	2,090806	2,389981	1,339942	2,225761	2,062582
	2	550000	2,302181	2,675473	1,662682	3,155992	2,820558
	2	625000	2,371518	2,45359	1,744957	2,637471	2,637557
	2	700000	1,977308	1,970997	1,575531	2,217626	2,445968
	2	775000	2,195613	2,180514	1,792859	2,417309	2,774038



	th	n	sse	paralelo	par-sse
L1	4	500	1,939145	0,004324	0,004304
	4	1000	2,569759	0,01146	0,011
	4	1500	2,525433	0,015597	0,016056
	4	2000	2,953169	0,024667	0,025084
	4	2500	2,683395	0,025926	0,029347
	4	3000	2,721146	0,024444	0,025227
	4	3500	2,675372	0,035544	0,037653
	4	4000	2,717019	0,033531	0,03516
L2	4	8000	2,261346	0,106805	0,096271
	4	13000	2,268554	0,158239	0,139499
	4	18000	2,889707	0,275546	0,295006
	4	23000	2,354592	0,265906	0,300159
	4	28000	2,298552	0,257116	0,280523
	4	33000	2,323157	0,39695	0,450536
	4	38000	2,370749	0,367843	0,260058
	4	43000	2,188808	0,415091	0,516637
	4	48000	2,356003	0,337102	0,39385
	4	53000	2,328433	0,341279	0,39053
	4	58000	2,35218	0,376247	0,435559
	4	100000	1,95254	0,443716	0,569002
	4	175000	2,344007	0,51299	0,614374
	4	250000	3,552328	0,78982	1,107131
	4	325000	2,41067	0,523104	0,754557
	4	400000	2,515103	0,493058	0,799112
4	475000	2,217462	0,517719	0,70436	
4	550000	2,026905	0,475604	0,623789	
4	625000	2,838799	0,604754	0,860168	
4	700000	2,564913	0,446343	0,682792	
RAM	4	775000	3,330452	0,785046	0,996018
	4	1600000	1,198099	2,021021	1,803207

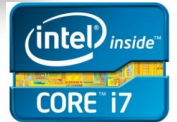


sswap

```

3 void sswap_seq(int n, float *x, float *y)
4 {
5     int i;
6     float temp;
7     for (i = 0; i < n; ++i) {
8         temp = x[i];
9         x[i] = y[i];
10        y[i] = temp;
11    }
12 }

```



TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	3.371534	6.090150	0.007084	0.007106	0.004067
4	1000	3.829753	6.977495	0.010808	0.014238	0.014236
4	1500	3.711646	7.251362	0.019805	0.020574	0.018798
4	2000	3.817765	7.631409	0.029697	0.028854	0.028948
4	2500	3.859603	7.565812	0.031776	0.034170	0.034210
4	3000	3.937442	7.582445	0.043129	0.043521	0.045405
4	3500	3.880370	7.568502	0.037928	0.050116	0.047185
4	4000	3.869755	7.514597	0.051699	0.056289	0.055118
4	8000	2.505626	2.560265	0.108536	0.104135	0.114210
4	13000	2.527251	2.559293	0.174296	0.177489	0.178722
4	18000	2.512554	2.563499	0.260676	0.250451	0.223495
4	23000	2.610880	2.647943	0.253542	0.328611	0.321779
4	28000	2.868175	2.993403	0.298976	0.408976	0.411720
4	33000	2.499488	2.532702	0.367455	0.389213	0.393380
4	38000	2.332522	2.401494	0.454027	0.468166	0.503888
4	43000	2.219003	2.341744	0.648490	0.598649	0.602189
4	48000	2.128420	2.180054	0.621071	0.612939	0.561868
4	53000	2.082740	2.060204	0.724686	0.651591	0.718728
4	58000	2.152618	2.206209	0.708574	0.727412	0.744731
4	100000	2.011606	2.071070	0.841696	1.137968	1.039329
4	175000	2.125320	2.136603	0.962528	1.841855	1.984916
4	250000	2.048598	2.094769	1.275962	1.977074	2.252364
4	325000	2.098520	2.140630	1.826282	2.974998	2.966890
4	400000	2.068795	2.072086	1.360187	2.438887	2.579294
4	475000	1.978652	2.071849	1.559945	2.630831	2.666863
4	550000	1.962228	1.907504	1.741101	2.910638	2.942876
4	625000	1.976808	1.933032	1.580201	2.837796	2.747562
4	700000	1.684886	1.881127	1.527633	3.473523	2.808752
4	775000	1.787477	1.821768	1.956924	3.030555	2.793813
4	1600000	1.836230	1.921820	3.103400	3.548694	3.622374

	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	3,312727	6,250429	0,00635	0,011276	0,009565
	2	1000	3,627907	6,723355	0,01695	0,029896	0,035999
	2	1500	3,462452	7,260431	0,027254	0,033237	0,033431
	2	2000	3,830979	6,913126	0,034031	0,025007	0,039798
	2	2500	3,870996	7,46678	0,030569	0,053861	0,053877
	2	3000	2,185866	6,854888	0,048675	0,034804	0,057663
	2	3500	3,448619	5,479014	0,092604	0,073383	0,086281
	2	4000	3,901931	7,515675	0,049519	0,069377	0,085852
L2	2	8000	2,522068	2,554557	0,140374	0,173742	0,175527
	2	13000	3,329303	4,044673	0,330593	0,441189	0,448243
	2	18000	3,394795	4,208919	0,390582	0,536104	0,656967
	2	23000	3,172806	4,203592	0,381735	0,513525	0,515814
	2	28000	2,922544	3,567051	0,477324	0,490614	0,608068
L3	2	33000	1,911673	2,621915	0,33602	0,597818	0,444801
	2	38000	3,031514	3,22582	0,493227	0,796747	0,66065
	2	43000	2,627394	3,164023	0,778227	0,731595	0,793529
	2	48000	3,028955	3,441994	0,700024	0,98864	1,007909
	2	53000	2,302147	2,692685	0,699241	0,822979	0,734071
	2	58000	2,266817	2,323674	0,561578	0,722045	0,725319
	2	100000	2,679654	2,650325	0,978628	1,226093	1,21705
	2	175000	2,343103	2,11969	0,959705	1,436289	1,509351
	2	250000	2,196738	2,416204	1,307848	1,905054	1,81947
RAM	2	325000	3,16474	3,148612	1,610235	2,721049	2,907674
	2	400000	1,841104	1,917136	0,99109	1,635686	1,508984
	2	475000	1,956253	1,993099	1,39229	2,094245	2,054095
	2	550000	2,208676	2,370669	1,991473	2,965453	2,728033
	2	625000	2,078668	2,106598	1,577202	2,295336	2,325339
	2	700000	2,10949	2,107161	1,903254	2,738509	2,52326
	2	775000	1,755816	1,894262	1,343773	2,757598	2,620567



	th	n	sse	paralelo	par-sse
L1	4	500	1,975	0,004173	0,004455
	4	1000	2,357069	0,00833	0,008206
	4	1500	2,529018	0,013193	0,012905
	4	2000	2,631148	0,021345	0,020963
	4	2500	2,682369	0,025469	0,025764
	4	3000	2,723665	0,038559	0,039747
	4	3500	3,055518	0,043951	0,047398
	4	4000	2,582699	0,043265	0,044386
L2	4	8000	2,33713	0,106478	0,095509
	4	13000	2,346395	0,138916	0,178469
	4	18000	2,242671	0,222558	0,277587
	4	23000	2,479776	0,0958	0,065388
	4	28000	2,296332	0,267592	0,288818
	4	33000	2,367314	0,381269	0,438468
	4	38000	2,35814	0,308434	0,337128
	4	43000	2,170186	0,325698	0,356497
	4	48000	2,348368	0,349447	0,405902
	4	53000	2,334518	0,365445	0,431873
	4	58000	2,359597	0,367812	0,441998
	4	100000	2,575376	0,672209	0,746851
	4	175000	2,406068	0,497371	0,564909
	4	250000	2,601191	0,651859	0,766777
	4	325000	2,375315	0,62546	0,748127
	4	400000	1,330995	0,584378	0,646799
4	475000	2,502346	0,558192	0,65247	
4	550000	2,123549	0,558274	0,677013	
4	625000	2,685228	0,648513	0,777389	
4	700000	0,995804	0,727648	0,758302	
RAM	4	775000	1,97274	0,620669	0,683978
	4	1600000	0,972466	1,354283	1,66914

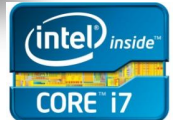


dswap

```

3 void dswap_seq(int n, double *x, double *y)
4 {
5     int i;
6     double temp;
7     for (i = 0; i < n; ++i){
8         temp = x[i];
9         x[i] = y[i];
10        y[i] = temp;
11    }
12 }

```



TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	1.959227	3.594488	0.007287	0.007096	0.006982
4	1000	1.965193	3.839180	0.017965	0.014851	0.020360
4	1500	1.984142	3.870087	0.008284	0.023594	0.024265
4	2000	1.983209	3.855003	0.023847	0.028772	0.010717
4	2500	1.339044	1.370070	0.040990	0.046876	0.042120
4	3000	1.341613	1.368554	0.040144	0.049262	0.047603
4	3500	1.351648	1.367165	0.066708	0.070945	0.065354
4	4000	1.547824	1.563505	0.081728	0.077577	0.073038
4	8000	1.297773	1.364604	0.112487	0.124936	0.120463
4	13000	2.418525	2.443570	0.359376	0.349723	0.315461
4	18000	1.258958	1.190913	0.280938	0.258959	0.257644
4	23000	1.242749	1.289746	0.333468	0.332866	0.370384
4	28000	1.168799	1.295633	0.405687	0.424767	0.422944
4	33000	1.244920	1.265643	0.444096	0.462491	0.487131
4	38000	1.299381	1.343149	0.602672	0.525452	0.619357
4	43000	1.238731	1.271210	0.621291	0.615582	0.644284
4	48000	1.205947	1.225917	0.604194	0.623216	0.611749
4	53000	1.181695	1.259381	0.692784	0.710355	0.722960
4	58000	1.272265	1.294437	0.724604	0.798617	0.809161
4	100000	1.226502	1.302713	1.143635	1.156686	1.176751
4	175000	1.229709	1.255652	1.243711	1.440911	1.418433
4	250000	1.213531	1.228336	1.398575	1.590798	1.346003
4	325000	1.147907	1.094847	1.407572	1.791352	1.794521
4	400000	1.174471	1.208928	1.488838	1.788112	1.940271
4	475000	1.118213	1.218744	1.763883	2.136689	1.794056
4	550000	1.061141	1.158202	1.581105	1.740585	2.078177
4	625000	0.964782	1.249598	1.716631	1.924265	2.169114
4	700000	0.953310	1.301554	1.783787	2.189477	2.306167
4	775000	1.126276	1.290591	2.127456	2.132637	2.478696
4	1600000	1.170473	1.174806	2.058144	2.163900	2.248324

	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	2,055926	4,248503	0,011298	0,01471	0,014497
	2	1000	1,96627	3,823656	0,011783	0,014333	0,014333
	2	1500	1,97506	3,856105	0,01785	0,021137	0,021146
	2	2000	1,976365	3,806976	0,023331	0,028315	0,028439
L2	2	2500	1,345281	1,381887	0,03394	0,040905	0,040818
	2	3000	1,346152	1,360185	0,058881	0,061613	0,07137
	2	3500	1,350603	1,369792	0,043276	0,055885	0,055399
	2	4000	1,355126	1,372615	0,075867	0,050183	0,062413
	2	8000	1,351673	1,37078	0,126453	0,097416	0,121375
	2	13000	1,351472	1,390641	0,179103	0,157001	0,196236
L3	2	18000	1,296622	1,344166	0,208219	0,268356	0,271476
	2	23000	1,20945	1,230771	0,267147	0,274753	0,30113
	2	28000	1,217122	1,241775	0,3101	0,322166	0,355402
	2	33000	1,213914	1,226564	0,367967	0,380066	0,393394
	2	38000	1,442379	1,464242	0,506572	0,600915	0,512819
	2	43000	1,223197	1,249076	0,478914	0,48611	0,485719
	2	48000	1,255471	1,272082	0,450512	0,549186	0,599639
	2	53000	1,477907	1,679209	0,654885	0,837859	0,846315
	2	58000	1,220219	1,23701	0,55072	0,624714	0,621332
	2	100000	1,21756	1,247732	0,810816	0,873713	0,894744
	2	175000	1,284037	1,317017	1,225803	1,51968	1,382934
RAM	2	250000	1,268488	1,266237	1,170257	1,552608	1,585518
	2	325000	1,171684	1,150828	0,966809	1,238815	1,40404
	2	400000	1,418396	1,448822	1,611507	1,704374	1,677423
	2	475000	1,113889	1,089721	1,206472	1,306341	1,33912
	2	550000	1,084134	1,129884	1,313199	1,443451	1,461457
	2	625000	1,064743	1,130845	1,322857	1,332754	1,446718
	2	700000	1,358127	1,370692	1,593732	1,734303	1,693101
	2	775000	1,15593	1,153419	1,370498	1,487985	1,511942



	th	n	sse	paralelo	par-sse
L1	4	500	1,808359	0,006799	0,007125
	4	1000	2,123288	0,015343	0,014593
	4	1500	1,853025	0,026639	0,027451
	4	2000	1,824533	0,027942	0,028684
L2	4	2500	1,829526	0,042593	0,042645
	4	3000	1,849657	0,059111	0,066971
	4	3500	1,846513	0,066069	0,039911
	4	4000	1,8593	0,069169	0,072735
	4	8000	1,880989	0,177626	0,212363
	4	13000	1,897242	0,227839	0,294728
	4	18000	1,908331	0,302582	0,337749
	4	23000	1,876197	0,252959	0,325208
	4	28000	1,914406	0,29401	0,34181
	4	33000	1,957291	0,332626	0,337009
	4	38000	1,936458	0,42056	0,572084
	4	43000	1,940805	0,364965	0,40683
	4	48000	2,442844	0,525343	0,617943
	4	53000	1,854964	0,423494	0,597813
	4	58000	2,345563	0,484849	0,553913
	4	100000	1,140051	0,424176	0,447361
4	175000	1,107795	0,580137	0,598298	
4	250000	2,187852	0,569015	0,620359	
4	325000	1,194631	0,528026	0,568256	
RAM	4	400000	1,765466	1,145475	1,520955
	4	475000	1,129863	1,372243	1,893551
	4	550000	0,990841	1,428072	1,429782
	4	625000	0,955979	1,447355	1,922968
	4	700000	0,961676	1,74224	1,520718
	4	775000	0,930645	1,61513	1,471384



axpy

iaxpy

```

3 void iaxpy_seq(int n, int a, int *x, int *y)
4 {
5     int i;
6     for (i = 0; i < n; ++i)
7         y[i] = a*x[i]+y[i];
8 }

```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	3.138960	5.655914	0.006232	0.006387	0.008604
4	1000	3.034962	5.041987	0.014463	0.014150	0.012694
4	1500	3.310484	5.919578	0.013006	0.027966	0.027548
4	2000	3.317916	6.338242	0.027073	0.028707	0.028069
4	2500	4.324566	8.130061	0.041923	0.044886	0.044145
4	3000	3.387492	6.402105	0.041737	0.028822	0.050243
4	3500	3.387018	6.513972	0.048692	0.049266	0.046386
4	4000	3.105572	6.467362	0.056068	0.018919	0.061951
4	8000	2.830533	3.294553	0.127760	0.107186	0.116540
4	13000	2.775973	3.489239	0.181843	0.211653	0.195761
4	18000	2.801734	3.605621	0.255118	0.253749	0.255300
4	23000	2.566795	3.514968	0.332523	0.305835	0.266418
4	28000	2.570849	3.277314	0.391673	0.368396	0.351384
4	33000	2.461994	2.705150	0.366071	0.430918	0.400560
4	38000	2.388239	2.677457	0.365887	0.438982	0.484846
4	43000	2.457752	2.842962	0.537959	0.501530	0.617915
4	48000	2.485851	2.990382	0.604153	0.588425	0.692733
4	53000	2.592947	2.831081	0.639376	0.606556	0.703306
4	58000	2.441782	2.921156	0.720826	0.705167	0.796943
4	100000	2.097423	2.554775	0.906088	1.201316	1.119310
4	175000	2.327534	2.446836	1.200573	1.752768	1.788499
4	250000	2.527628	2.822664	1.426717	2.256946	2.494580
4	325000	2.321447	2.426782	1.595173	2.263627	2.458649
4	400000	2.318670	2.391418	1.565922	3.375220	3.404729
4	475000	2.303435	2.349349	1.521975	2.720220	3.028714
4	550000	2.090155	2.411522	1.652159	3.104599	3.131451
4	625000	2.273994	2.002382	1.416058	3.139333	3.786664
4	700000	2.172303	2.233573	1.510817	2.857253	3.098030
4	775000	1.878321	2.080268	1.651770	2.652089	3.208705
4	1600000	1.765495	1.779885	1.639207	2.793516	2.639490



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	3,069514	5,536254	0,006072	0,007595	0,007664
	2	1000	3,059746	5,063815	0,013072	0,016968	0,02155
	2	1500	3,103421	5,821313	0,024643	0,057246	0,039868
	2	2000	3,156832	6,370188	0,037386	0,041606	0,043066
	2	2500	4,083839	4,489158	0,035333	0,043492	0,043595
	2	3000	3,431441	6,179721	0,055667	0,062563	0,039032
	2	3500	3,35031	5,923297	0,064954	0,042693	0,060372
	2	4000	4,410341	4,421302	0,09359	0,109173	0,096403
L2	2	8000	2,605338	2,950523	0,069089	0,150778	0,167316
	2	13000	2,3788	3,020266	0,151761	0,239499	0,15863
	2	18000	3,131473	5,197269	0,384568	0,53165	0,671532
	2	23000	2,684413	4,528927	0,379254	0,34278	0,579567
	2	28000	2,610138	3,112863	0,268344	0,363132	0,485533
L3	2	33000	2,561112	3,050935	0,388528	0,520714	0,404439
	2	38000	1,810959	2,407609	0,378518	0,621463	0,651909
	2	43000	2,165881	2,388789	0,493817	0,60376	0,56571
	2	48000	1,959161	3,17567	0,563257	0,723643	0,569363
	2	53000	2,273614	2,909904	0,557183	0,704286	0,644398
	2	58000	1,643378	2,437752	0,581171	0,799579	0,769268
	2	100000	2,200729	2,590721	0,73056	0,888387	1,114905
	2	175000	1,986219	2,729181	0,977028	1,793528	1,900578
	2	250000	2,890947	4,438194	1,722776	2,793054	2,848558
RAM	2	325000	2,208004	2,350315	1,076091	1,638598	1,8189
	2	400000	2,225765	2,360613	1,246066	1,906927	2,039725
	2	475000	2,23693	2,303727	1,257197	1,966383	1,979515
	2	550000	2,01629	2,088254	1,251681	1,676649	2,089431
	2	625000	1,910844	1,92339	1,130074	1,942135	1,65011
	2	700000	2,917293	3,004126	1,830318	2,861749	3,241672
	2	775000	2,766964	2,614851	1,332422	2,624416	2,489565



	th	n	sse	paralelo	par-sse
L1	4	500	1,853831	0,007606	0,008116
	4	1000	2,16417	0,011764	0,011698
	4	1500	2,256837	0,010805	0,014894
	4	2000	2,294545	0,018123	0,017705
	4	2500	2,396499	0,024316	0,02421
	4	3000	2,419173	0,036736	0,031218
	4	3500	2,480739	0,047152	0,041709
	4	4000	2,379792	0,037843	0,036592
L2	4	8000	2,036846	0,096831	0,088399
	4	13000	2,066359	0,155087	0,142658
	4	18000	2,089868	0,176268	0,179808
	4	23000	2,40834	0,294427	0,304132
	4	28000	2,059237	0,260123	0,278681
	4	33000	2,1147	0,397051	0,441502
	4	38000	2,239325	0,429428	0,45563
	4	43000	2,039268	0,338223	0,371834
	4	48000	2,103027	0,393031	0,554686
	4	53000	2,058683	0,362876	0,423245
	4	58000	2,112259	0,361217	0,363197
	4	100000	2,101599	0,509682	0,583227
	4	175000	2,117924	0,64791	0,837655
	4	250000	1,876012	0,68544	0,831512
	4	325000	1,460216	0,6776	0,880914
	4	400000	1,400972	0,757202	0,889376
4	475000	1,204834	0,771567	0,89097	
4	550000	1,708417	1,122925	1,369621	
4	625000	0,868484	0,817082	0,956773	
4	700000	0,895052	1,010261	1,228025	
RAM	4	775000	1,12222	1,248753	1,167928
	4	1600000	0,788287	1,541253	1,702704



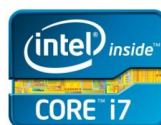
saxpy

```

3 void saxpy_seq(int n, float a, float *x, float *y)
4 {
5     int i;
6     for (i = 0; i < n; ++i)
7         y[i] = a*x[i]+y[i];
8 }

```

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	3.576288	6.062603	6.312178	0.007498	0.007416	0.010777	0.010820
4	1000	3.411905	5.950997	6.335102	0.017586	0.019039	0.018058	0.017209
4	1500	6.653014	13.178760	13.266268	0.041226	0.040276	0.040206	0.038363
4	2000	3.851035	7.407543	7.426996	0.030173	0.006711	0.027805	0.019076
4	2500	3.921508	7.384551	7.366097	0.037845	0.039953	0.038594	0.039351
4	3000	3.942490	7.584318	7.622526	0.047542	0.045545	0.051349	0.047353
4	3500	3.902547	7.585486	7.682342	0.053143	0.056997	0.056653	0.057327
4	4000	3.860215	7.471286	2.512936	0.048061	0.056233	0.039780	0.049940
4	8000	2.831294	3.072308	3.068234	0.109340	0.119421	0.125626	0.110809
4	13000	3.000649	3.406002	3.380520	0.183139	0.186623	0.168737	0.178057
4	18000	3.053318	3.513367	3.502847	0.277241	0.230705	0.238921	0.225120
4	23000	2.891756	3.335553	3.363028	0.313727	0.289317	0.317980	0.404954
4	28000	2.724041	3.133128	3.117115	0.294782	0.342314	0.388711	0.393368
4	33000	2.798868	3.010794	3.179151	0.327548	0.443266	0.438944	0.448975
4	38000	2.388118	2.575393	2.591576	0.423548	0.535917	0.457156	0.434625
4	43000	2.789482	3.130293	3.114701	0.547469	0.588258	0.618263	0.615084
4	48000	2.554068	2.721332	2.712594	0.557556	0.521307	0.425034	0.686159
4	53000	2.574475	2.689611	2.750101	0.591329	0.561987	0.630998	0.686191
4	58000	2.975657	3.276720	3.208216	0.782420	0.914592	0.772575	0.743620
4	100000	2.164882	2.394022	2.413550	0.845713	1.081701	1.047792	1.067620
4	175000	2.793776	2.902671	2.883505	1.308967	2.121300	2.223630	2.171225
4	250000	2.435407	2.569648	2.554842	1.118140	1.947859	2.053535	0.395483
4	325000	2.319982	2.417456	2.400034	1.376175	2.459367	2.447678	2.444579
4	400000	2.343044	1.749163	2.361075	1.461517	2.593670	2.896228	2.355990
4	475000	2.443886	2.526135	2.505522	1.774469	3.175569	3.067886	2.910725
4	550000	3.130081	2.993391	2.873331	1.916471	3.492293	2.659368	4.190927
4	625000	2.392140	2.697657	2.829993	1.803816	3.846623	3.665556	3.682741
4	700000	1.785939	1.626708	1.964800	1.708977	3.038862	2.872965	2.773879
4	775000	1.605662	1.759241	2.107361	1.821027	3.064414	3.679651	4.067634
4	1600000	1.928259	1.843286	1.904007	2.531070	3.347168	3.738216	3.556301



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma
L1	2	500	3,593933	5,858054	6,225424	0,008947	0,010855	0,009585	0,009678
	2	1000	3,47093	7,341532	6,476331	0,032287	0,034028	0,023531	0,04088
	2	1500	3,808596	7,086323	7,174801	0,039613	0,042528	0,056496	0,056731
	2	2000	3,831235	4,835024	4,384427	0,030184	0,038571	0,043469	0,043573
	2	2500	3,853029	7,007255	7,392729	0,019769	0,080661	0,049034	0,086731
	2	3000	3,910162	7,22449	7,681736	0,053091	0,049505	0,064671	0,057685
	2	3500	3,838203	6,643501	7,149905	0,098587	0,09832	0,071342	0,131485
	2	4000	3,907405	7,466543	7,510242	0,075549	0,07026	0,06578	0,072204
L2	2	8000	3,446677	5,01316	4,884599	0,132473	0,204912	0,208489	0,292735
	2	13000	2,462128	3,043349	2,867716	0,162015	0,183285	0,18608	0,186237
	2	18000	3,32462	4,768234	4,805398	0,23927	0,404397	0,428968	0,4249
	2	23000	3,230987	4,688673	4,431687	0,469081	0,697216	0,723816	0,719696
	2	28000	3,371136	4,663166	4,951251	0,406478	0,967859	1,001255	1,00353
L3	2	33000	2,752689	2,582217	2,217298	0,394488	0,458605	0,469613	0,471983
	2	38000	3,39415	4,435064	5,099894	0,823818	0,764491	1,060792	1,259391
	2	43000	3,14492	4,650795	4,43652	0,798567	1,159943	0,909051	0,785528
	2	48000	3,022529	3,738793	3,951523	0,610108	1,239345	1,283126	1,290586
	2	53000	1,761166	2,153891	2,693956	0,507582	0,760796	0,781711	0,742341
	2	58000	4,03318	4,557199	4,563554	0,671318	0,381733	0,953065	1,164106
	2	100000	3,020203	3,895312	3,970166	1,066293	1,977895	1,662813	1,641814
	2	175000	1,853115	1,96533	1,924354	0,720329	1,464895	1,428867	1,579351
	2	250000	3,117274	3,87064	4,410517	1,802666	2,793951	2,82899	2,835047
	2	325000	3,441095	3,775535	3,864875	2,192616	3,685566	3,507222	2,799502
RAM	2	400000	2,342462	2,556777	2,497869	1,512689	2,25555	2,0243	2,505159
	2	475000	3,017322	3,768753	3,785754	2,334286	3,900793	4,026145	3,824336
	2	550000	3,15128	3,163932	3,096608	1,796305	2,937407	3,073688	3,074921
	2	625000	2,51408	2,839382	2,770727	1,906015	2,912808	3,03343	2,943326
	2	700000	1,982868	2,004225	1,996135	1,312273	2,014373	2,100548	2,197954
	2	775000	2,467659	2,433326	2,483966	1,387114	2,418637	2,25633	2,011255



	th	n	sse	paralelo	par-sse
L1	4	500	2,021705	0,005509	0,005437
	4	1000	2,474368	0,011574	0,013317
	4	1500	2,650934	0,022727	0,023692
	4	2000	2,730359	0,02358	0,024372
	4	2500	2,813543	0,029135	0,032966
	4	3000	2,861037	0,029653	0,030442
	4	3500	2,907459	0,040141	0,042175
	4	4000	2,816601	0,043977	0,047946
L2	4	8000	2,356601	0,041957	0,048546
	4	13000	2,197173	0,129728	0,122965
	4	18000	2,225746	0,179019	0,184942
	4	23000	2,239602	0,247673	0,2588
	4	28000	2,15358	0,25713	0,268972
	4	33000	2,259089	0,317847	0,325635
	4	38000	2,260986	0,358024	0,46543
	4	43000	2,784079	0,459927	0,580361
	4	48000	2,220798	0,343682	0,362723
	4	53000	2,272941	0,371926	0,413094
	4	58000	2,265926	0,439801	0,606873
	4	100000	2,288774	0,516001	0,571217
	4	175000	2,85961	0,573964	0,970687
	4	250000	2,256898	0,640448	0,797279
	4	325000	1,741555	0,731726	0,770394
	4	400000	1,797613	0,598472	0,76374
4	475000	2,110709	0,557419	0,714175	
4	550000	2,576886	0,632327	0,834199	
4	625000	1,012434	0,765525	0,937354	
4	700000	1,445857	0,748619	0,780304	
RAM	4	775000	1,193545	1,289943	1,546743



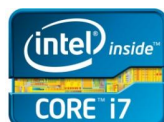
daxpy

```

3 void daxpy_seq(int n, double a, double *x, double *y)
4 {
5     int i;
6     for (i = 0; i < n; ++i)
7         y[i] = a*x[i]+y[i];
8 }

```

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	2.026842	2.681085	3.633421	0.008124	0.008282	0.007353	0.007667
4	1000	1.952199	3.791512	3.757624	0.013891	0.014519	0.006406	0.012798
4	1500	1.987668	3.775018	3.821121	0.003891	0.024359	0.025750	0.029355
4	2000	1.922070	3.795792	3.365572	0.036314	0.037433	0.037460	0.033421
4	2500	1.257690	1.298618	1.294396	0.037204	0.037289	0.046293	0.032600
4	3000	1.545990	1.608456	1.617257	0.044134	0.048054	0.047829	0.043540
4	3500	1.592739	1.674074	1.689815	0.056721	0.061603	0.052937	0.057726
4	4000	1.446231	1.117355	1.546617	0.059188	0.059291	0.059739	0.050744
4	8000	1.646969	1.860151	1.856902	0.115622	0.117898	0.101915	0.107962
4	13000	1.428391	1.572483	1.563339	0.052709	0.170510	0.166216	0.177835
4	18000	1.557972	1.649957	1.707210	0.311031	0.271802	0.300228	0.299011
4	23000	1.387211	1.535726	1.526426	0.269465	0.320303	0.317723	0.347554
4	28000	1.422023	1.524750	1.520411	0.329038	0.371361	0.400252	0.434241
4	33000	1.444501	1.062802	1.519505	0.391696	0.410017	0.436184	0.364429
4	38000	1.442395	1.300580	1.529863	0.541991	0.489279	0.642376	0.602905
4	43000	1.463156	1.562831	1.544922	0.516745	0.563159	0.622999	0.658656
4	48000	1.211624	1.270920	1.232792	0.535144	0.606033	0.648869	0.641814
4	53000	1.318469	1.284509	1.279933	0.576148	0.640462	0.657622	0.660577
4	58000	1.239789	1.306491	1.295142	0.525761	0.631761	0.840185	0.663316
4	100000	1.264885	1.337287	1.335143	0.801393	0.918433	1.081600	1.029256
4	175000	1.057341	1.431656	1.413816	1.320172	1.445542	1.322508	1.455001
4	250000	1.217561	1.257522	1.286290	1.234195	1.411107	1.369657	1.533531
4	325000	1.217069	1.040938	1.171644	1.428921	1.615477	1.429639	1.749905
4	400000	1.210598	1.080810	1.271756	1.581395	1.806479	2.165227	2.314390
4	475000	1.015879	1.157614	1.121813	1.688128	1.790424	2.447112	2.373812
4	550000	1.068729	1.294479	1.252486	1.713872	2.267605	2.199973	2.251964
4	625000	0.936667	1.221005	1.191482	2.070239	2.273885	2.167120	2.204581
4	700000	1.032178	1.256372	1.207220	2.018080	2.114987	2.268606	2.048216
4	775000	1.124168	1.253017	1.327224	2.092019	2.211081	2.378517	2.022032
4	1600000	1.160132	1.192572	1.181358	1.723994	2.191257	2.176265	2.283171



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma
L1	2	500	2,211938	3,751991	4,258231	0,017163	0,014727	0,014893	0,014801
	2	1000	1,873367	3,608763	3,621141	0,031697	0,034059	0,020838	0,023329
	2	1500	2,65524	3,107254	3,03718	0,022739	0,034651	0,041332	0,04545
	2	2000	1,913369	3,629691	3,273683	0,049504	0,08383	0,084057	0,059024
L2	2	2500	1,260233	1,285395	1,316083	0,048693	0,04904	0,055257	0,055199
	2	3000	2,140305	2,22924	2,229098	0,101365	0,10678	0,08234	0,072762
	2	3500	1,360417	1,438599	1,441045	0,052691	0,049721	0,054055	0,054551
	2	4000	1,893718	2,729808	2,816272	0,09083	0,088582	0,115782	0,115648
	2	8000	1,457834	1,618705	1,51148	0,09691	0,155188	0,156822	0,159048
	2	13000	1,794204	2,271806	2,914018	0,325048	0,365327	0,376342	0,37797
L3	2	18000	1,275465	1,22359	1,263225	0,274362	0,316764	0,325285	0,328765
	2	23000	1,591614	2,316856	2,352662	0,488577	0,597956	0,421552	0,618588
	2	28000	1,317911	1,055884	1,200653	0,328809	0,372951	0,381264	0,343719
	2	33000	1,565201	1,849387	1,895741	0,514251	0,742807	0,795065	0,796157
	2	38000	1,243325	1,549484	1,55287	0,4004	0,49922	0,643601	0,587679
	2	43000	2,035966	2,210807	2,292578	0,709442	0,829513	0,897576	0,911911
	2	48000	1,087945	1,228828	1,229753	0,382799	0,536368	0,574752	0,637353
	2	53000	1,181627	1,397804	1,411396	0,622436	0,751186	0,77052	0,692334
	2	58000	1,272416	1,457374	1,455963	0,650938	0,754548	0,836808	0,792192
	2	100000	1,116985	1,523951	1,564472	0,766656	0,964989	1,01565	1,011536
RAM	2	175000	1,229663	1,174573	1,32667	0,91775	1,098237	1,133269	1,082047
	2	250000	0,942665	1,106994	1,057458	0,770065	0,944192	0,6066	1,295104
	2	325000	1,517235	1,932719	1,892908	1,761786	1,889155	1,862281	1,462616
	2	400000	1,210537	1,31947	1,524939	1,465898	1,62606	1,664698	1,306584
	2	475000	1,122394	1,205954	1,216088	1,006213	1,061586	1,025718	1,097883
	2	550000	1,173193	1,14362	1,134837	1,092752	1,338226	1,375968	1,461771
	2	625000	1,270734	1,267895	1,434471	1,648163	1,877859	1,890025	1,873071
	2	700000	1,543436	1,542041	1,523746	1,677014	1,762801	1,684856	1,829375
2	775000	0,76149	1,167911	1,302404	1,478055	1,336403	1,570277	1,468312	



	th	n	sse	paralelo	par-sse
L1	4	500	1,282524	0,004732	0,004705
	4	1000	1,417412	0,01161	0,012012
	4	1500	1,477229	0,02096	0,021186
	4	2000	1,417913	0,021404	0,021053
L2	4	2500	1,345144	0,037799	0,039006
	4	3000	1,359234	0,035472	0,036808
	4	3500	1,359831	0,048289	0,050315
	4	4000	1,346307	0,057648	0,057462
	4	8000	1,381636	0,092084	0,112399
	4	13000	1,384933	0,18979	0,217947
	4	18000	1,385589	0,186855	0,194331
	4	23000	1,388545	0,235346	0,240936
	4	28000	1,383172	0,243724	0,268672
	4	33000	1,385208	0,284452	0,290349
	4	38000	1,38412	0,303688	0,310139
	4	43000	1,390035	0,288359	0,304639
	4	48000	1,389443	0,334826	0,354674
	4	53000	1,363426	0,349222	0,365518
	4	58000	1,384994	0,30536	0,360928
	4	100000	1,392931	0,373669	0,407067
4	175000	1,828985	0,519507	0,530467	
4	250000	1,728939	0,629717	0,551694	
4	325000	1,449641	0,631238	0,729582	
RAM	4	400000	1,723479	0,990639	1,138555
	4	475000	1,074811	1,85256	2,030822
	4	550000	1,074834	1,71289	1,759319
	4	625000	0,996039	1,547366	1,876402
	4	700000	0,97438	1,898426	1,742603
	4	775000	0,789363	1,553055	1,593749



rot

irrot

```

8 void irot_seq(int n, int *x, int *y, int c, int s)
9 {
10     int i;
11     int temp;
12
13     for (i = 0; i < n; ++i){
14         temp = c*x[i] + s*y[i];
15         y[i] = c*y[i] - s*x[i];
16         x[i] = temp;
17     }
18 }
    
```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	2.358947	4.407915	0.017036	0.016815	0.014553
4	1000	2.371289	4.642292	0.030217	0.041237	0.039597
4	1500	3.290244	6.407286	0.073577	0.081583	0.065567
4	2000	1.804386	4.745817	0.081452	0.112692	0.086234
4	2500	2.447956	2.704879	0.091150	0.067651	0.098013
4	3000	2.401209	4.764154	0.061018	0.117666	0.094768
4	3500	2.407518	4.767910	0.119938	0.127156	0.131232
4	4000	2.400687	4.765082	0.103698	0.055494	0.139001
4	8000	2.419313	4.780122	0.236143	0.239634	0.198975
4	13000	2.423331	4.804102	0.344522	0.390847	0.384410
4	18000	2.409540	4.792891	0.488488	0.529274	0.523006
4	23000	2.417419	2.645042	0.667320	0.656082	0.674237
4	28000	2.474266	4.933552	0.638557	0.743010	0.835652
4	33000	2.547811	5.064816	0.761854	0.873909	0.950962
4	38000	2.429419	4.669132	0.817081	0.930011	1.057919
4	43000	2.468625	4.398090	0.916723	1.440324	1.255519
4	48000	2.404610	4.376427	0.906887	1.228150	1.213624
4	53000	2.428338	4.417648	0.977620	1.547732	1.481391
4	58000	2.419894	4.329473	1.020946	1.361211	1.456995
4	100000	2.421617	4.236087	1.271237	2.036801	2.473077
4	175000	2.437386	4.440527	1.505692	2.666266	3.576218
4	250000	2.433375	4.191484	1.384183	3.060659	4.779066
4	325000	2.483171	4.519592	1.731496	3.693455	5.229090
4	400000	2.448807	3.712135	1.839442	3.773685	5.504987
4	475000	2.441460	4.349061	1.961595	3.527525	5.592494
4	550000	2.334524	4.560269	2.049940	4.232716	6.475527
4	625000	2.414731	3.978191	1.784725	4.121867	5.626771
4	700000	1.934783	4.144444	1.815962	4.448493	5.993147
4	775000	1.846192	3.875582	2.105354	4.438588	7.383967
4	1600000	2.730560	4.097475	3.605360	7.018471	6.159290



	th	n	sse	avx	paralelo	par-sse	par-avx
L1	2	500	2,35227	4,442505	0,013225	0,022208	0,027511
	2	1000	2,105945	4,175838	0,02806	0,031911	0,041899
	2	1500	2,386482	4,689782	0,037831	0,054216	0,054051
	2	2000	2,15692	4,748778	0,077309	0,059194	0,105041
	2	2500	2,816824	6,143406	0,089444	0,119959	0,118221
	2	3000	2,749495	5,865469	0,101866	0,133477	0,144164
	2	3500	2,875982	5,517388	0,10698	0,167013	0,165605
	2	4000	2,951383	5,892365	0,214438	0,152793	0,264758
L2	2	8000	2,268827	4,414191	0,242751	0,317334	0,250566
	2	13000	2,906175	6,125544	0,557269	0,439498	0,584248
	2	18000	2,544745	4,72014	0,511676	0,633725	0,515463
	2	23000	3,247284	6,432721	0,695284	0,668446	0,977561
	2	28000	2,527198	5,720449	0,550718	0,704294	0,963171
L3	2	33000	2,388189	4,753413	0,642544	0,908222	1,217597
	2	38000	2,853186	5,503798	0,711582	1,02028	1,241727
	2	43000	3,00289	5,460432	0,868579	1,430258	1,885125
	2	48000	2,808301	5,050012	0,69145	1,081759	1,213116
	2	53000	2,639803	4,606828	0,770905	1,203606	1,504483
	2	58000	2,632589	4,707901	0,723831	1,148425	1,44488
	2	100000	2,78265	6,220074	1,605644	2,365679	2,767826
	2	175000	2,650854	4,902944	1,472167	2,22214	3,073433
	2	250000	3,197193	5,690936	1,534855	2,421246	4,637853
RAM	2	325000	2,451584	4,271267	1,638114	3,31077	3,897333
	2	400000	2,461575	4,091478	1,591494	3,051728	3,808313
	2	475000	2,454426	4,141008	1,639807	3,329939	4,240847
	2	550000	3,21243	5,297938	2,085265	4,264578	5,702823
	2	625000	2,570973	4,328277	1,941338	4,106011	5,251753
	2	700000	2,359501	3,911285	1,806692	3,467877	4,389078
	2	775000	2,394152	3,319016	1,751115	3,226273	4,110044



	th	n	sse	paralelo	par-sse
L1	4	500	2,912688	0,016087	0,016307
	4	1000	3,021779	0,030332	0,030017
	4	1500	3,07999	0,044219	0,043028
	4	2000	3,155878	0,07416	0,083694
	4	2500	3,160111	0,080289	0,080605
	4	3000	3,193473	0,095621	0,108793
	4	3500	3,180541	0,1314	0,119735
	4	4000	3,205303	0,123958	0,125376
L2	4	8000	3,292022	0,301529	0,322694
	4	13000	3,343392	0,363879	0,398056
	4	18000	3,37702	0,483764	0,572069
	4	23000	3,346612	0,544335	0,714306
	4	28000	3,338887	0,814905	1,155364
	4	33000	3,33589	0,615084	0,895923
	4	38000	3,600465	1,006075	1,349216
	4	43000	4,176659	0,839005	1,286477
	4	48000	3,364813	0,776081	1,125878
	4	53000	3,337034	0,987464	1,633679
	4	58000	3,344594	0,87899	0,96403
	4	100000	3,34531	0,990899	1,50955
	4	175000	3,893861	1,299549	2,188113
	4	250000	3,319306	1,184447	1,970012
	4	325000	3,41226	1,256845	2,102043
	4	400000	3,549146	1,450124	2,28589
	4	475000	3,644198	1,292228	2,225337
4	550000	4,031911	1,560172	2,566677	
4	625000	1,646028	1,152982	1,998104	
4	700000	2,032716	1,219146	1,686162	
RAM	4	775000	3,229363	1,116588	1,599991



srot

```

9 void srot_seq(int n, float *x, float *y, float c, float s)
10 {
11     int i;
12     float temp;
13
14     for (i = 0; i < n; ++i){
15         temp = c*x[i] + s*y[i];
16         y[i] = c*y[i] - s*x[i];
17         x[i] = temp;
18     }
19 }

```

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	4.644560	8.677638	8.464589	0.017639	0.017287	0.016289	0.019593
4	1000	4.905529	9.830178	9.114610	0.041947	0.043730	0.043019	0.016583
4	1500	3.917321	10.667685	6.952234	0.062672	0.069257	0.017178	0.076412
4	2000	5.239353	11.139752	10.028804	0.065936	0.070488	0.066210	0.083475
4	2500	5.049851	10.516706	9.464132	0.078093	0.083676	0.082952	0.081282
4	3000	5.230813	11.100854	9.913565	0.052707	0.095820	0.091168	0.027010
4	3500	5.046040	10.682071	9.628249	0.122177	0.124558	0.138695	0.115626
4	4000	5.029445	10.230399	9.542069	0.131289	0.128492	0.135750	0.114293
4	8000	3.999749	6.137815	5.956579	0.295877	0.260439	0.250157	0.250793
4	13000	4.294121	6.199433	6.004825	0.440986	0.424032	0.376834	0.519832
4	18000	4.983552	6.059771	5.902849	0.470359	0.604691	0.575942	0.534601
4	23000	3.709750	6.076575	5.901332	0.720208	0.727787	0.678069	0.745413
4	28000	5.012329	6.098659	5.925083	0.568798	1.010345	0.772028	0.173214
4	33000	4.965977	5.709898	5.353703	0.646233	0.840290	0.938556	0.974772
4	38000	4.696395	5.185802	5.172253	0.707583	1.369844	1.047620	1.073271
4	43000	4.039970	4.649932	4.554643	0.877336	1.179362	1.328617	1.372701
4	48000	3.981594	4.647305	4.582499	0.962171	1.301015	1.634183	1.558943
4	53000	4.296684	4.557658	4.374089	0.954431	1.551677	1.726051	1.450310
4	58000	4.318063	4.603861	4.398162	1.013601	1.594941	1.681035	1.608822
4	100000	4.262952	4.509266	4.626235	1.303334	2.197628	2.463652	2.699122
4	175000	4.333132	4.646954	4.596950	1.521216	3.568890	3.841446	3.658689
4	250000	4.275819	4.587421	4.577759	1.381643	3.677355	4.397124	4.548093
4	325000	4.301929	4.598837	4.539981	1.686259	2.592203	4.196873	4.488803
4	400000	4.059383	3.544515	3.693502	1.381556	4.694313	6.078302	5.877464
4	475000	3.752476	4.628587	4.313346	1.750806	5.133644	5.371002	6.333122
4	550000	3.261192	4.205360	4.361391	1.524350	5.835136	6.129138	7.314771
4	625000	3.360028	4.052312	4.406473	1.971192	5.584219	7.583365	8.788606
4	700000	3.662422	4.360073	4.318989	1.896881	6.421198	8.572539	8.460137
4	775000	3.626938	3.546223	3.970263	1.837892	7.122291	8.860080	8.283661
4	1600000	3.372103	3.733617	3.732820	3.238119	6.310973	5.439848	6.513570



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma
L1	2	500	4,676225	8,648987	8,281625	0,013437	0,016874	0,018797	0,019633
	2	1000	4,926091	9,101129	8,219286	0,026847	0,030071	0,036896	0,036387
	2	1500	6,396849	11,967719	11,515192	0,081872	0,053047	0,056058	0,068719
	2	2000	6,005725	12,715934	11,485432	0,168955	0,19746	0,198744	0,16254
	2	2500	6,162153	11,00578	11,012698	0,221964	0,177107	0,191146	0,127738
	2	3000	5,103458	10,744284	9,038468	0,121359	0,156881	0,159261	0,121933
	2	3500	5,063788	10,401686	9,585392	0,127154	0,137956	0,14047	0,152206
	2	4000	5,126186	10,002041	9,695351	0,272595	0,403954	0,411598	0,388889
L2	2	8000	2,571952	10,115993	9,675399	0,497084	0,753332	0,767574	0,602513
	2	13000	5,022756	9,538477	9,904064	0,633098	1,13327	1,185724	1,1852
	2	18000	3,920056	6,471797	7,222368	0,578751	0,538815	0,677292	0,659561
	2	23000	5,821046	9,525125	8,502751	0,772824	0,940928	1,137755	1,086287
	2	28000	3,605837	6,329531	6,777901	0,648563	0,747782	0,975419	0,945503
L3	2	33000	3,118977	4,827823	4,794354	0,690732	0,838737	1,015991	1,057569
	2	38000	3,757109	6,554846	6,431533	0,787368	0,99565	1,388315	1,400347
	2	43000	4,949464	7,513808	7,307141	1,345371	2,10245	1,558034	2,159292
	2	48000	4,93115	7,59163	7,44866	1,416207	2,223086	2,353408	2,359228
	2	53000	5,80028	6,178666	6,170241	1,021756	1,658785	1,795142	1,789404
	2	58000	5,53426	5,899579	5,89671	0,952478	1,522362	1,636804	1,651824
	2	100000	4,56736	5,019797	5,067919	0,908499	2,592315	2,698126	2,712596
	2	175000	4,373043	4,631023	4,666334	1,343411	2,656904	3,021343	3,047463
	2	250000	5,403567	5,673538	5,863247	1,823575	4,142437	4,759737	4,62106
	2	325000	7,325609	6,109311	7,963058	2,864238	7,305899	6,080303	8,074587
RAM	2	400000	5,414196	5,999331	5,952243	2,078678	5,721947	6,567845	6,673364
	2	475000	3,058135	4,329895	4,291503	1,560607	3,796812	4,607705	4,733857
	2	550000	3,556469	2,663325	4,706025	1,387312	4,338274	5,494981	4,914184
	2	625000	3,964096	4,199006	4,03972	1,764715	4,098101	4,747693	4,961339
	2	700000	3,754587	5,15413	4,773492	2,314239	5,509451	6,521492	6,192414
	2	775000	3,915112	4,001781	4,113178	1,793928	4,949926	4,713836	5,058823



	th	n	sse	paralelo	par-sse
L1	4	500	4,454264	0,018392	0,019205
	4	1000	5,085804	0,027584	0,030332
	4	1500	5,317035	0,056805	0,055938
	4	2000	5,445056	0,075607	0,07551
	4	2500	5,532267	0,071334	0,074753
	4	3000	5,564337	0,104236	0,104161
	4	3500	5,539169	0,126078	0,13064
	4	4000	5,536868	0,119639	0,11805
L2	4	8000	5,017459	0,234822	0,220711
	4	13000	4,952642	0,470064	0,460736
	4	18000	5,095461	0,576515	0,628712
	4	23000	5,105979	0,647902	0,790763
	4	28000	5,117216	0,643833	0,771556
	4	33000	5,969241	0,996727	1,172097
	4	33000	6,069341	1,056727	1,196097
	4	43000	6,496119	1,142713	1,332463
	4	48000	5,122355	0,825778	1,039931
	4	53000	5,068049	0,86287	1,054337
	4	58000	7,04994	1,20971	1,496531
	4	100000	5,086982	1,093013	1,553106
	4	175000	4,863306	1,004954	1,675096
	4	250000	7,115261	1,809421	2,555236
	4	325000	5,412891	1,467228	1,900636
	4	400000	5,415832	1,240455	1,875248
4	475000	4,988181	1,246753	1,777037	
4	550000	5,380591	1,347323	1,630836	
4	625000	2,760383	1,346174	1,291782	
4	700000	3,443585	1,279962	1,469563	
RAM	4	775000	3,232685	1,15687	1,360022
	4	1600000	0,851007	1,286002	1,61428



drot

```

9 void drot_seq(int n, double *x, double *y, double c, double s)
10 {
11     int i;
12     double temp;
13
14     for (i = 0; i < n; ++i){
15         temp = c*x[i] + s*y[i];
16         y[i] = c*y[i] - s*x[i];
17         x[i] = temp;
18     }
19 }

```

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	2.520280	5.073198	5.824176	0.016855	0.017197	0.013160	0.020705
4	1000	2.533437	4.519504	6.282746	0.038085	0.042416	0.049455	0.048905
4	1500	2.549169	5.424624	6.354117	0.051791	0.053426	0.052845	0.045449
4	2000	2.539823	5.208174	6.044692	0.065496	0.068111	0.071461	0.069552
4	2500	2.445830	2.740635	2.789606	0.069535	0.100892	0.101980	0.094525
4	3000	2.467487	3.045285	3.015724	0.097527	0.101098	0.104020	0.103784
4	3500	2.465330	2.992063	2.982681	0.129565	0.131836	0.105091	0.111287
4	4000	2.060430	3.139404	3.137943	0.131675	0.138690	0.131859	0.124459
4	8000	2.466433	3.033627	3.013142	0.230403	0.249928	0.240618	0.248786
4	13000	2.502720	3.044232	3.021494	0.429877	0.336700	0.378313	0.384146
4	18000	2.236844	2.626315	2.630723	0.483340	0.481033	0.539979	0.562883
4	23000	2.159615	2.296968	2.307796	0.618977	0.560688	0.672801	0.703239
4	28000	2.349028	2.537279	2.553512	0.722991	0.825846	0.818940	0.811510
4	33000	2.168189	2.303561	2.315822	0.734007	0.847146	0.825310	0.954911
4	38000	2.025067	2.292267	2.307254	0.800463	0.998747	0.941364	0.814271
4	43000	2.338045	2.536956	2.515116	1.000452	1.221338	1.196515	1.342195
4	48000	2.103972	2.579058	2.582999	0.954737	1.308429	1.345937	1.288376
4	53000	2.102719	2.280114	2.199943	0.938892	1.241189	1.303081	1.322875
4	58000	2.369635	2.537378	2.559301	1.058527	1.460333	1.603913	1.549621
4	100000	2.170598	2.291999	2.322745	1.268166	1.761496	2.038499	2.189604
4	175000	2.177476	2.289035	2.305106	1.682294	2.552474	2.689371	2.716269
4	250000	2.037597	2.203338	2.222785	1.703022	2.771397	2.991091	2.997709
4	325000	2.057765	2.229302	2.097952	1.723206	2.789788	3.311948	3.154492
4	400000	1.598938	2.100377	2.175001	1.897145	3.332822	4.373098	4.163257
4	475000	1.588621	1.862039	1.875904	2.208821	3.637532	4.382019	4.163566
4	550000	1.729626	1.832614	2.066118	2.938564	3.856921	4.207389	3.802461
4	625000	2.060601	2.079839	2.193109	3.030876	4.072367	4.253784	4.088130
4	700000	1.779626	1.888922	1.980016	1.961202	3.177664	3.937973	4.046375
4	775000	1.798551	1.953254	1.862649	2.231444	3.826381	3.770247	3.771093
4	1600000	1.741406	1.815816	1.732284	2.791095	3.201492	2.871488	3.373293



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma
L1	2	500	2,526036	5,040782	5,840129	0,014113	0,027245	0,028028	0,026666
	2	1000	2,54175	5,37669	6,320268	0,05106	0,0472	0,052201	0,05306
	2	1500	2,555811	3,125366	6,170483	0,041094	0,0544	0,053218	0,052927
	2	2000	2,54389	5,201081	4,731898	0,0837	0,095078	0,106559	0,105894
L2	2	2500	2,292945	3,868166	3,958903	0,099022	0,162234	0,177822	0,186317
	2	3000	5,003644	6,132286	6,077656	0,187375	0,289795	0,311438	0,30738
	2	3500	2,270438	3,330768	3,408511	0,161106	0,115752	0,178697	0,206313
	2	4000	2,587583	4,742197	5,150512	0,175828	0,258207	0,253802	0,247321
	2	8000	2,404949	4,817732	4,796352	0,462136	0,572998	0,534592	0,46995
	2	13000	2,564679	4,083413	5,865036	0,760468	1,007285	1,070771	1,070664
L3	2	18000	2,284572	2,595563	2,316261	0,460275	0,642506	0,686636	0,689411
	2	23000	2,171202	3,299466	3,296235	0,679168	0,97739	1,048324	1,052701
	2	28000	1,369163	2,001393	2,076416	0,549786	0,606321	0,630333	0,640399
	2	33000	2,773323	3,035917	3,043656	0,763949	0,901386	0,970828	0,989014
	2	38000	3,287382	3,747911	3,388145	1,050206	1,398847	1,34916	1,350119
	2	43000	3,071322	4,33173	4,366464	1,318913	1,574267	1,727545	1,7026
	2	48000	2,557498	2,898546	2,895634	1,020184	1,15969	1,253259	1,268433
	2	53000	2,361092	2,741947	2,962812	1,058011	1,324544	1,433301	1,440173
	2	58000	2,321624	2,480392	2,468987	0,881458	1,107781	1,217636	1,18752
	2	100000	2,169181	3,743151	3,782585	1,737698	2,37506	2,691728	2,669102
	2	175000	2,877982	3,063959	3,03628	1,977256	3,238634	3,318755	3,090438
RAM	2	250000	2,319563	2,484699	2,502716	1,628915	2,333306	2,680453	2,578382
	2	325000	2,277611	2,42813	2,236766	1,344613	2,295628	2,791266	2,70629
	2	400000	1,827743	1,919512	1,900618	1,00865	1,81298	1,933477	2,330413
	2	475000	1,564248	2,120531	2,110548	1,875663	2,263566	2,487869	2,507633
	2	550000	2,230508	2,304941	2,316692	1,780468	2,559137	2,828349	2,512806
	2	625000	1,840678	1,968283	1,83764	0,920422	2,09824	1,946523	2,376695
	2	700000	1,72972	1,803558	1,752332	1,444237	1,939814	2,232356	2,296974
	2	775000	1,500912	1,731	1,780564	1,377778	2,035648	2,163627	2,017406



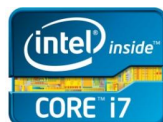
	th	n	sse	paralelo	par-sse
L1	4	500	2,530675	0,017114	0,018463
	4	1000	2,694301	0,038574	0,038024
	4	1500	2,758421	0,056335	0,050047
	4	2000	2,773295	0,077122	0,068407
L2	4	2500	2,472772	0,075051	0,074421
	4	3000	2,308355	0,077578	0,131352
	4	3500	2,497492	0,133338	0,118241
	4	4000	2,435667	0,147319	0,137745
	4	8000	2,553656	0,285487	0,361179
	4	13000	2,507254	0,445642	0,426032
	4	18000	2,543225	0,455965	0,511943
	4	23000	2,552988	0,637955	0,733178
	4	28000	2,5393	0,64051	0,77371
	4	33000	2,546437	0,596802	0,58055
	4	38000	2,54977	0,726402	0,95863
	4	43000	2,547092	0,943919	1,017906
	4	48000	2,364069	0,863281	1,044781
	4	53000	2,559269	0,904558	1,103119
	4	58000	2,526795	0,638098	0,686825
	4	100000	2,494341	0,660334	0,900331
4	175000	2,984152	0,766821	0,985303	
4	250000	3,041881	0,878181	0,934514	
4	325000	2,356994	0,645823	0,894134	
RAM	4	400000	1,818948	1,058318	1,271761
	4	475000	0,852254	1,356323	1,875306
	4	550000	1,020756	1,663919	1,946376
	4	625000	0,949583	1,938925	1,812915
	4	700000	0,91012	2,054066	2,064518
	4	775000	0,890787	1,790526	1,790584



rotm**irotm**

```
4 void irotm_seq(int n, int *x, int *y, int *param)
5 {
6     int flag, h11, h12, h21, h22, two, w, z, zero;
7     flag = param[0];
8     int i;
9
10    if(flag+2 == 0) return;
11    if(flag<0){
12        h11 = param[1];
13        h12 = param[3];
14        h21 = param[2];
15        h22 = param[4];
16        for(i=0;i<n;i++){
17            w = x[i];
18            z = y[i];
19            x[i] = w*h11 + z*h12;
20            y[i] = w*h21 + z*h22;
21        }
22    }else if(flag == 0){
23        h12 = param[3];
24        h21 = param[2];
25        for(i=0;i<n;i++){
26            w = x[i];
27            z = y[i];
28            x[i] = w + z*h12;
29            y[i] = w*h21 + z;
30        }
31    }else{
32        h11 = param[1];
33        h22 = param[4];
34        for(i=0;i<n;i++){
35            w = x[i];
36            z = y[i];
37            x[i] = w*h11 + z;
38            y[i] = -w + z*h22;
39        }
40    }
41 }
```

TH	N	sse	avx	omp	omp-sse	omp-avx
4	500	1.922045	3.422071	0.010760	0.019405	0.019851
4	1000	1.194021	2.632475	0.030357	0.033834	0.033308
4	1500	1.347562	2.653715	0.034154	0.036034	0.042254
4	2000	1.374005	2.657570	0.054781	0.019658	0.056992
4	2500	1.358345	2.691597	0.063173	0.060829	0.061152
4	3000	1.341244	2.708684	0.065997	0.072552	0.067695
4	3500	1.381405	2.741120	0.076015	0.072287	0.099443
4	4000	1.380823	2.759147	0.094429	0.114036	0.130488
4	8000	1.431282	2.837852	0.174956	0.186925	0.193280
4	13000	1.396020	2.787618	0.231463	0.331495	0.299735
4	18000	1.383450	2.760317	0.347846	0.376979	0.394174
4	23000	1.341906	2.501430	0.482116	0.437164	0.507284
4	28000	1.781925	3.622768	0.635946	0.614433	0.717606
4	33000	1.445598	2.944271	0.343978	0.654815	0.282218
4	38000	1.518357	2.788274	0.591134	0.725935	0.935244
4	43000	1.424577	2.764091	0.668207	0.888130	0.887990
4	48000	1.403086	2.756444	0.471851	1.014722	0.840123
4	53000	1.454191	2.844277	0.783911	0.950181	1.036579
4	58000	1.443750	2.794589	0.714579	0.924721	1.180530
4	100000	1.370335	2.726258	1.079512	1.416694	1.686784
4	175000	1.391239	2.754950	1.327010	1.700608	2.443950
4	250000	1.413021	2.821476	1.511168	2.504431	3.223066
4	325000	1.610710	3.247813	1.874347	3.075106	3.926229
4	400000	1.344063	2.757520	1.611057	2.076790	3.622451
4	475000	1.412335	2.763932	1.388015	2.292309	3.742693
4	550000	1.377018	2.762481	1.750855	3.216968	4.149343
4	625000	1.395412	2.197277	1.781416	3.383836	4.115239
4	700000	1.319034	2.592011	1.498505	2.212045	3.876905
4	775000	1.361856	2.291771	1.774657	3.118492	4.321133



	th	n	Sse	avx	paralelo	par-sse	par-avx
L1	2	500	1,349956	2,606812	0,014262	0,018609	0,014549
	2	1000	1,345217	2,61617	0,019925	0,026316	0,025721
	2	1500	1,527715	3,008876	0,038975	0,044407	0,042159
	2	2000	1,380977	2,723276	0,046703	0,044721	0,06234
	2	2500	1,355195	2,678843	0,067731	0,054131	0,056023
	2	3000	1,320356	2,703284	0,062174	0,06129	0,075104
	2	3500	1,288224	2,685389	0,086229	0,062805	0,085977
	2	4000	1,369236	2,737775	0,112946	0,119352	0,106327
L2	2	8000	1,283299	2,728712	0,179673	0,188807	0,144433
	2	13000	1,371998	2,726401	0,235727	0,297387	0,418797
	2	18000	1,378032	2,72553	0,283314	0,424694	0,320868
	2	23000	1,366968	2,744216	0,322601	0,58722	0,382552
	2	28000	1,307372	2,790727	0,488841	0,678342	0,445176
L3	2	33000	1,380177	2,775251	0,520151	0,714278	0,523018
	2	38000	1,402412	2,803641	0,465655	0,623899	0,638407
	2	43000	1,416533	2,773071	0,588546	0,626385	0,691338
	2	48000	1,392559	2,75471	0,603835	0,829996	0,689096
	2	53000	1,376466	2,697568	0,643971	0,69605	0,789536
	2	58000	2,200216	4,227053	1,068111	1,704892	1,446636
	2	100000	1,408128	2,771569	0,87086	1,073713	1,367686
	2	175000	1,334308	2,725513	1,244637	1,488935	1,947062
	2	250000	1,58553	3,165262	1,508374	1,905158	2,695155
	2	325000	1,438585	2,820838	1,359441	1,947831	2,633553
RAM	2	400000	1,379778	2,692385	1,437851	1,96746	2,473332
	2	475000	1,430827	2,778879	1,484653	2,070785	2,401113
	2	550000	1,440866	2,808508	1,59937	2,166966	3,184598
	2	625000	1,405893	2,604447	1,572342	2,136942	3,025546
	2	700000	1,397838	2,62058	1,572112	2,26974	2,718376
	2	775000	1,380623	2,313423	1,494034	2,202037	2,707482

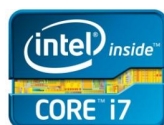


	th	n	sse	paralelo	par-sse
L1	4	500	2,004699	0,010858	0,011926
	4	1000	2,201038	0,019515	0,018903
	4	1500	2,281259	0,026431	0,025885
	4	2000	2,365054	0,039613	0,039284
	4	2500	2,229392	0,06212	0,055415
	4	3000	2,360592	0,072942	0,078327
	4	3500	2,349705	0,085492	0,083788
	4	4000	2,393388	0,096065	0,095786
L2	4	8000	2,484413	0,170317	0,159021
	4	13000	2,486747	0,34682	0,381545
	4	18000	2,509776	0,267378	0,314968
	4	23000	2,503768	0,379793	0,445515
	4	28000	2,51022	0,429243	0,510249
	4	33000	2,505746	0,552474	0,762079
	4	38000	2,513075	0,493087	0,605682
	4	43000	2,490151	0,545498	0,641902
	4	48000	2,514296	0,582736	0,700181
	4	53000	2,52268	0,542508	0,723461
	4	58000	2,513752	0,669558	0,911177
	4	100000	3,033797	0,998515	1,189961
	4	175000	2,507879	0,700615	1,095234
	4	250000	2,73533	0,881852	1,370324
	4	325000	2,751147	0,867686	1,38224
	4	400000	1,759384	0,893558	1,31921
4	475000	2,479004	0,902847	1,299969	
4	550000	2,3538	0,87102	1,253043	
4	625000	2,465767	0,746059	1,135524	
4	700000	3,073882	0,799117	1,26244	
RAM	4	775000	1,311365	1,03985	1,628736
	4	1600000	0,883638	1,708224	1,605209



srotm

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	3.219641	6.153380	6.715859	0.012436	0.012063	0.011926	0.010731
4	1000	3.495902	7.207001	7.877309	0.025497	0.024800	0.024897	0.022883
4	1500	3.541361	7.262815	8.019766	0.039191	0.052228	0.040336	0.035843
4	2000	3.552076	7.825355	8.572514	0.051664	0.053486	0.052847	0.026531
4	2500	3.598737	7.617481	8.411013	0.065636	0.072556	0.065683	0.057627
4	3000	3.622859	7.943886	8.749815	0.075947	0.072167	0.068905	0.069303
4	3500	3.656302	7.912594	8.797283	0.073147	0.075102	0.096259	0.097227
4	4000	3.620243	7.263198	8.538406	0.090150	0.094719	0.096310	0.092343
4	8000	3.520249	4.058501	4.030759	0.198473	0.181961	0.183630	0.175210
4	13000	3.653278	4.315176	4.266186	0.187432	0.274064	0.289568	0.390746
4	18000	3.774202	4.344170	4.321841	0.350233	0.438698	0.466779	0.414223
4	23000	3.218960	4.383129	4.381430	0.469233	0.522981	0.481906	0.518262
4	28000	3.870938	4.351603	4.379363	0.507038	0.610675	0.647713	0.608931
4	33000	3.730715	4.283468	4.302240	0.596975	0.658186	0.744133	0.396313
4	38000	2.907755	3.660458	3.772261	0.695968	0.887364	0.755371	0.818882
4	43000	5.416934	6.022798	6.127081	1.299535	1.532224	1.579111	1.592795
4	48000	2.981392	3.428512	3.608225	0.886416	1.119352	1.118258	1.096643
4	53000	2.946315	3.084288	3.137502	0.756063	0.987040	1.201723	1.051546
4	58000	2.887242	3.599553	3.629512	0.903014	1.202512	1.231677	1.237808
4	100000	2.894740	3.105971	3.120946	1.143230	1.669584	1.725946	1.774799
4	175000	3.223000	3.321703	2.938451	1.338160	2.432512	2.937772	2.820265
4	250000	3.085574	3.249449	3.252700	1.763196	3.380701	3.711324	3.463538
4	325000	2.858238	3.130080	2.534543	1.200223	1.929475	3.484006	3.373693
4	400000	2.939542	3.231124	3.369271	1.685421	3.783922	4.179458	3.856563
4	475000	2.765137	2.271990	2.778850	1.554794	3.758094	4.127521	4.121707
4	550000	2.256233	3.502929	4.209803	2.640227	6.314082	7.132277	7.014363
4	625000	2.285048	2.328070	2.819023	1.756144	3.934125	4.367912	5.290265
4	700000	2.110510	2.771098	2.666029	1.443625	3.776561	5.300754	5.613734
4	775000	2.383659	2.891468	2.901615	1.716497	4.517987	5.732837	4.224005



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma
L1	2	500	3,521562	6,005211	6,712621	0,020382	0,022664	0,022556	0,022511
	2	1000	3,480803	7,085258	7,641762	0,027794	0,033137	0,029215	0,020139
	2	1500	4,605387	9,280111	10,286781	0,040218	0,049478	0,048313	0,058399
	2	2000	3,717459	5,883696	5,643701	0,023321	0,045643	0,098256	0,059951
	2	2500	3,625287	7,600253	8,671577	0,079617	0,076735	0,075108	0,065361
	2	3000	3,879665	7,742706	9,309863	0,069838	0,08208	0,065531	0,074789
	2	3500	3,590338	7,773571	8,645403	0,110191	0,122552	0,121191	0,101602
	2	4000	3,594107	7,48234	8,48739	0,117289	0,097903	0,132283	0,105412
L2	2	8000	5,745582	6,473909	6,600398	0,29071	0,298616	0,308397	0,257266
	2	13000	3,645237	4,081986	4,153194	0,270694	0,373043	0,354059	0,297827
	2	18000	3,634916	4,137797	3,98237	0,302554	0,46062	0,526872	0,348299
	2	23000	3,80753	5,675665	6,185718	0,517439	0,894032	1,033253	1,170445
	2	28000	3,356039	5,696913	5,919472	0,707893	1,111784	0,950273	0,870816
L3	2	33000	2,117634	3,34121	3,368343	0,557741	0,699985	0,708964	0,754414
	2	38000	2,102162	3,053676	3,465643	0,53717	0,760716	0,73109	0,658126
	2	43000	2,117885	3,132513	3,109539	0,536186	0,949021	0,866131	0,881708
	2	48000	1,959707	3,008429	2,802439	0,620591	0,98952	0,805836	0,924334
	2	53000	3,677095	3,865601	3,851574	0,790276	1,150529	0,988474	1,181
	2	58000	3,351357	4,296103	4,678517	0,999481	1,793942	1,996818	1,992043
	2	100000	2,88895	4,099475	4,173779	1,300382	2,008239	2,07731	2,082084
	2	175000	3,287564	3,565271	3,407667	1,255785	2,828492	2,260202	2,299616
	2	250000	3,121538	3,422537	3,521953	1,291234	2,54713	2,829909	2,80032
	2	325000	3,41163	3,67097	3,669547	1,57634	2,966636	3,064974	3,327269
RAM	2	400000	2,77303	3,24349	2,98788	1,128216	2,793567	3,366391	3,33057
	2	475000	2,575499	2,855217	2,991464	1,502168	2,820556	3,202884	3,35542
	2	550000	2,702698	2,948599	2,931324	1,641809	3,069165	3,393075	3,453095
	2	625000	2,97172	3,215188	3,088158	1,885428	3,466117	3,01043	3,126918
	2	700000	2,291397	2,679663	2,878559	1,541313	2,956523	3,260105	3,764377
	2	775000	2,843461	2,900831	2,826982	1,748624	3,38497	3,418792	3,654731

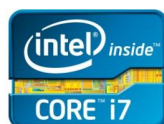


	th	n	sse	paralelo	par-sse
L1	4	500	2,508021	0,010271	0,009952
	4	1000	2,565732	0,015144	0,01538
	4	1500	2,972525	0,024563	0,024363
	4	2000	3,019271	0,042358	0,041352
	4	2500	3,064997	0,051988	0,05383
	4	3000	3,097325	0,068591	0,081707
	4	3500	3,111068	0,070802	0,074107
	4	4000	3,049795	0,078754	0,085843
L2	4	8000	2,546711	0,143966	0,158046
	4	13000	2,464648	0,199516	0,219966
	4	18000	2,574327	0,263329	0,261481
	4	23000	2,633851	0,304919	0,344749
	4	28000	2,613779	0,448836	0,566623
	4	33000	2,537221	0,369483	0,44107
	4	38000	2,597152	0,475027	0,693213
	4	43000	2,401851	0,520844	0,584721
	4	48000	2,613247	0,421405	0,564079
	4	53000	2,536265	0,497495	0,56023
	4	58000	3,287988	0,769692	1,144249
	4	100000	2,836049	0,652178	0,848742
	4	175000	2,603725	0,633269	0,937069
	4	250000	2,578742	0,727583	0,936668
	4	325000	2,968482	0,884997	1,067677
	4	400000	2,730887	0,783894	1,043047
4	475000	2,624506	0,689271	0,995882	
4	550000	3,325432	0,854091	1,062407	
4	625000	2,632618	0,622221	0,725129	
4	700000	3,023734	0,7834	0,801637	
RAM	4	775000	1,457552	0,693857	0,779566
	4	1600000	0,926345	1,57193	1,786224



drotm

TH	N	sse	avx	fma	omp	omp-sse	omp-avx	omp-fma
4	500	2.114519	3.681899	4.919905	0.012591	0.012549	0.014552	0.017813
4	1000	2.137737	4.345714	5.258427	0.028568	0.022085	0.024373	0.023196
4	1500	2.160729	4.367775	5.362212	0.048928	0.037799	0.036991	0.040677
4	2000	1.655047	4.352105	5.268387	0.048292	0.046802	0.052516	0.060175
4	2500	1.844583	2.409721	2.439319	0.061244	0.067895	0.079945	0.084477
4	3000	1.852940	2.403696	2.332609	0.080584	0.079356	0.070988	0.081850
4	3500	1.844522	2.401986	2.394217	0.083222	0.098749	0.091093	0.088108
4	4000	1.925453	2.084507	2.067801	0.086899	0.086706	0.083252	0.121491
4	8000	1.934581	2.165825	2.161484	0.171135	0.171863	0.178366	0.184978
4	13000	1.871448	2.422911	2.394209	0.259986	0.325911	0.311671	0.306135
4	18000	1.741525	2.136985	2.165546	0.364542	0.432699	0.451384	0.482532
4	23000	1.505934	1.628185	1.620381	0.488467	0.495584	0.446937	0.476172
4	28000	1.699103	1.775971	1.776600	0.519553	0.530201	0.580667	0.598969
4	33000	1.529300	1.603553	1.609741	0.545018	0.642542	0.586216	0.585345
4	38000	1.614168	1.797574	1.800823	0.702486	0.743068	0.805897	0.790172
4	43000	1.803833	1.894059	1.909581	0.760562	0.864164	0.921195	0.900792
4	48000	1.490331	1.655292	1.668450	0.671115	0.847059	0.866063	0.985234
4	53000	1.505580	1.580434	1.593045	0.718134	0.860177	0.908326	0.921656
4	58000	2.769554	2.984462	2.873702	0.903192	1.488855	1.755772	1.769646
4	100000	1.507942	1.581014	1.514496	0.991096	1.299196	1.408670	1.382684
4	175000	1.529206	1.456119	1.606763	0.795203	1.905313	1.966607	1.968297
4	250000	1.436461	1.491759	1.573057	1.522100	1.920492	1.938357	2.125035
4	325000	1.426732	1.657796	1.656280	1.542513	2.839859	2.920742	2.702823
4	400000	1.288202	1.538877	1.545593	1.911504	2.932797	2.925062	2.996724
4	475000	1.414013	1.456249	1.460375	2.104321	2.473101	2.730310	2.856714
4	550000	1.322405	1.446564	1.518323	2.096683	2.708385	2.991984	2.595888
4	625000	1.278787	1.453743	1.442435	2.257112	2.849829	2.547825	2.027643
4	700000	1.321757	1.310686	1.384096	2.123238	2.642006	2.116548	2.557671
4	775000	1.363641	1.432385	1.350874	2.304766	1.725356	2.280198	2.375376



	th	n	sse	avx	fma	paralelo	par-sse	par-avx	par-fma
L1	2	500	2,276829	7,676375	9,272869	0,034248	0,035741	0,036433	0,036267
	2	1000	2,138894	4,353467	5,40417	0,034679	0,027475	0,024664	0,025643
	2	1500	2,159486	4,38897	5,377593	0,05074	0,052533	0,053872	0,05352
	2	2000	2,245064	4,420465	4,374709	0,026019	0,04869	0,125303	0,133815
L2	2	2500	2,035986	2,151085	2,252414	0,085623	0,094006	0,091671	0,094205
	2	3000	2,465704	4,250805	4,695285	0,042654	0,203048	0,180104	0,101619
	2	3500	2,018952	2,191348	2,174294	0,113132	0,127542	0,127602	0,128803
	2	4000	1,927078	4,121438	4,520647	0,193988	0,20538	0,220119	0,200418
	2	8000	1,874659	2,123575	2,14561	0,209544	0,256001	0,188306	0,224169
	2	13000	2,260228	3,288661	3,249489	0,468395	0,609579	0,604223	0,271544
L3	2	18000	1,605932	1,810303	1,796104	0,272465	0,413326	0,448838	0,475812
	2	23000	2,332096	3,443908	3,406646	0,377876	0,401133	1,101999	1,106976
	2	28000	1,563903	1,62821	1,614337	0,414687	0,511853	0,543251	0,527561
	2	33000	0,952534	1,701031	1,709605	0,444948	0,624729	0,689202	0,702876
	2	38000	1,608523	1,696947	1,639643	0,436667	0,594914	0,615403	0,639977
	2	43000	2,130732	2,653333	2,377639	0,656241	1,04242	0,965468	0,146652
	2	48000	1,535586	1,615207	1,556321	0,552881	0,788578	0,836218	0,844702
	2	53000	1,41318	2,616266	2,716153	1,021783	1,357764	1,375377	1,526051
	2	58000	1,485163	1,600043	1,595413	0,617959	0,736343	0,792783	0,783062
	2	100000	1,663695	1,742867	1,738629	0,917821	1,126031	1,258681	1,162861
RAM	2	175000	1,53388	1,592523	1,609819	1,094479	1,41446	1,277768	1,54683
	2	250000	1,239246	1,44381	1,552313	1,261381	1,556736	1,710364	1,669047
	2	325000	1,371006	1,50834	1,520372	1,387451	1,604961	1,674495	1,614802
	2	400000	1,375016	1,499426	1,511323	1,087776	1,580886	1,86986	1,814617
	2	475000	1,290857	1,372799	1,39444	1,375695	1,588891	1,339971	1,748476
	2	550000	1,470359	1,510999	1,546294	1,387159	1,942537	1,927929	1,656799
	2	625000	1,223988	1,264426	1,272024	1,393185	1,585983	1,413042	1,343378
	2	700000	1,197162	1,281699	1,301249	1,229025	1,560152	1,382186	1,22115
	2	775000	1,328096	1,370163	1,362541	1,405347	1,722809	1,367843	0,69019



	th	n	sse	paralelo	par-sse
L1	4	500	1,583661	0,009655	0,013024
	4	1000	1,722596	0,023145	0,023363
	4	1500	1,757302	0,035885	0,035901
	4	2000	1,752179	0,04796	0,046254
L2	4	2500	1,61667	0,058234	0,05269
	4	3000	1,619249	0,066335	0,067982
	4	3500	1,628519	0,061121	0,062789
	4	4000	1,586226	0,100273	0,103887
	4	8000	1,176325	0,136338	0,159725
	4	13000	1,658154	0,247786	0,271782
	4	18000	1,659462	0,266732	0,273307
	4	23000	1,655736	0,315319	0,310092
	4	28000	2,224299	0,456711	0,478822
	4	33000	1,636838	0,381856	0,355524
	4	38000	1,950124	0,551887	0,59883
	4	43000	1,658857	0,407221	0,451314
	4	48000	1,414552	0,384915	0,47045
	4	53000	1,65371	0,418463	0,481265
	4	58000	1,660856	0,444428	0,485163
	4	100000	1,952963	0,526078	0,663225
4	175000	1,29826	0,608489	0,6141	
4	250000	1,725463	0,615267	0,542942	
4	325000	1,649161	0,590953	0,488761	
RAM	4	400000	1,650405	1,080458	1,176655
	4	475000	1,19927	1,274813	2,163225
	4	550000	1,292517	2,028567	2,577576
	4	625000	1,188269	2,147485	1,981902
	4	700000	1,123433	1,81551	2,371699
	4	775000	0,980271	2,142568	2,045784



