

Iscpu:	CPU(s): 4
	Nombre del modelo: Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz
	Virtualización: VT-x
	Caché L3: 3072 K



Prácticas de Estructura de Computadores
por Javier Fernández y Mancia Anguita
licencia [BY-NC-SA](#)

Raúl Ruano Narváez

POPCOUNT:	for i in 0 g 1 2; do printf "__OPTIM%1c__%48s\n" \$i "" tr " " "=" rm popcount gcc popcount.c -o popcount -O\$1 -D TEST=0 for j in \$(seq 0 10); do echo \$j; ./popcount done pr -11 -l 22 -w 80 done
	ignorar medición 0, repetir columna si alguna medición se sale demasiado de la media

Optimización -O0	0	1	2	3	4	5	6	7	8	9	10	media
popcount1 (lenguaje C - for):	106428	102331	112542	104329	99570	103805	101966	103520	103512	100841	103282	103570
popcount2 (lenguaje C - while):	53887	67319	56633	55179	52117	52161	59186	52339	53624	50037	63926	56252
popcount3 (leng.ASM-body while 4i):	15920	17101	17280	15008	15515	14930	15928	16281	17100	16239	18586	16397
popcount4 (leng.ASM-body while 3i):	14533	14845	14192	16344	14807	13853	14316	14645	14776	14415	13751	14594
popcount5 (CS:APP2e 3.49-group 8b):	26477	27393	25358	25684	26969	25456	26877	31634	26852	25836	26074	26813
popcount6 (Wikipedia- naive - 32b):	10994	12771	10249	10377	10681	10344	10947	15144	10104	10558	10343	11152
popcount7 (Wikipedia- naive -128b):	6435	7813	6043	7813	6108	5995	6424	6566	6199	6383	6449	6629
popcount8 (asm SSE3 - pshufb 128b):	1162	1183	1065	1077	1066	1400	1107	1117	1122	1170	1088	1145
popcount9 (asm SSE4- popcount 32b):	2539	2440	2195	2271	2384	2268	2248	2334	2242	2381	2197	2298
popcount10 (asm SSE4- popcount128b):	1169	1155	1215	1080	1255	1086	1160	1324	1075	1090	1059	1138

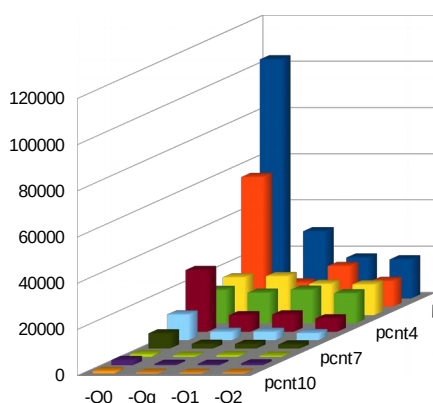
Optimización -Og	0	1	2	3	4	5	6	7	8	9	10	media
popcount1 (lenguaje C - for):	36433	29018	29287	26958	31630	27464	35305	30845	27399	26943	26846	29170
popcount2 (lenguaje C - while):	15257	10208	10493	10353	10585	11005	10592	11055	10828	10573	10195	10589
popcount3 (leng.ASM-body while 4i):	19835	15998	16052	16421	16342	18976	16032	17478	16327	19654	15900	16918
popcount4 (leng.ASM-body while 3i):	15566	12378	12514	12367	12327	14281	12226	16874	12559	12878	15203	13361
popcount5 (CS:APP2e 3.49-group 8b):	7086	7305	6724	6763	6937	8820	6760	6692	6685	7983	6737	7254
popcount6 (Wikipedia- naive - 32b):	3490	3700	3391	3428	3668	4025	3650	3595	3446	3618	3747	3653
popcount7 (Wikipedia- naive -128b):	2009	2156	2169	2050	2047	2227	2025	1984	1999	2016	1991	2085
popcount8 (asm SSE3 - pshufb 128b):	548	594	529	620	545	649	539	581	523	539	517	570
popcount9 (asm SSE4- popcount 32b):	651	649	603	605	612	670	613	614	635	611	678	633
popcount10 (asm SSE4- popcount128b):	424	441	441	412	414	521	415	419	408	414	410	436

Optimización -O1	0	1	2	3	4	5	6	7	8	9	10	media
popcount1 (lenguaje C - for):	25668	20012	19196	18378	16680	16493	17669	16601	17501	17033	18151	17771
popcount2 (lenguaje C - while):	25524	23610	16678	16199	17597	17587	16442	16004	16876	18889	17288	17717
popcount3 (leng.ASM-body while 4i):	13332	13264	15340	13090	13665	13859	13075	13095	13307	13259	13931	13589
popcount4 (leng.ASM-body while 3i):	14811	13187	15849	12707	15602	13621	14677	15780	15475	13767	16062	14673
popcount5 (CS:APP2e 3.49-group 8b):	7570	9090	7124	7339	7255	7098	7701	7088	7823	7299	7721	7578
popcount6 (Wikipedia- naive - 32b):	3591	4054	3407	3502	3426	4162	3684	3437	3689	3506	3685	3678
popcount7 (Wikipedia- naive -128b):	1897	1970	1911	1852	2062	2617	1825	1904	2060	2006	1977	2028
popcount8 (asm SSE3 - pshufb 128b):	524	554	614	527	537	1849	711	620	587	559	566	740
popcount9 (asm SSE4- popcount 32b):	599	612	678	661	604	703	675	673	703	602	607	638
popcount10 (asm SSE4- popcount128b):	404	453	420	415	407	450	455	416	456	411	405	423

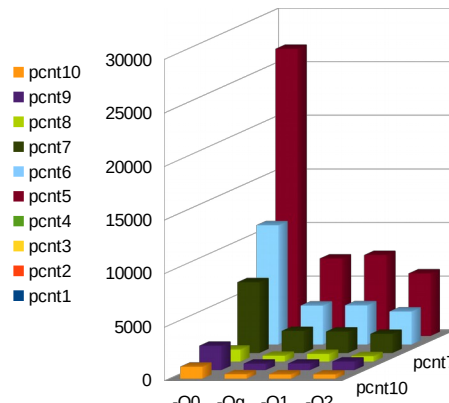
Optimización -O2	0	1	2	3	4	5	6	7	8	9	10	media
popcount1 (lenguaje C - for):	19070	16628	17209	17090	16852	16641	16614	17258	16623	17156	17660	16973
popcount2 (lenguaje C - while):	10106	10077	11959	10276	10074	10108	11763	11753	10260	15957	10283	11251
popcount3 (leng.ASM-body while 4i):	12878	13024	12966	13514	13749	13501	13454	13629	12718	13843	12966	13336
popcount4 (leng.ASM-body while 3i):	12234	14219	12289	13617	12048	13383	13211	12366	14178	13561	12433	13131
popcount5 (CS:APP2e 3.49-group 8b):	5521	6954	5553	5733	5653	5865	5931	5788	6243	5497	5665	5856
popcount6 (Wikipedia- naive - 32b):	3030	3063	3168	3057	3007	3038	3230	3124	3223	3046	3218	3103
popcount7 (Wikipedia- naive -128b):	1774	1882	1768	1734	1739	1762	1862	1727	2081	1982	1772	1813
popcount8 (asm SSE3 - pshufb 128b):	586	579	507	501	509	564	566	526	623	554	513	532
popcount9 (asm SSE4- popcount 32b):	807	836	788	820	834	857	898	842	1063	820	830	826
popcount10 (asm SSE4- popcount128b):	417	423	404	412	410	440	526	447	917	405	409	415

POPCOUNT:	-O0	-Og	-O1	-O2	Ganancias:	-O0	-Og	-O1	-O2	Comentario
pcnt1	103570	29170	17771	16973	pcnt1	-	-	1,00	-	comparado con el for más rápido
pcnt2	56252	10589	17717	11251	pcnt2	-	1,68	-	-	el while es un 70% más rápido
pcnt3	16397	16918	13589	13336	pcnt3	-	-	1,31	-	ASM se queda en un 35%
pcnt4	14594	13361	14673	13131	pcnt4	-	-	1,21	-	o en un 43%
pcnt5	26813	7254	7578	5856	pcnt5	-	-	-	3,03	sumar en grupos 8b sale 3x más rápido
pcnt6	11152	3653	3678	3103	pcnt6	-	-	-	5,73	sumar en árbol 6x
pcnt7	6629	2085	2028	1813	pcnt7	-	-	-	9,80	lectura 128b sube a 10x
pcnt8	1145	570	740	532	pcnt8	-	-	-	33,38	SSSE3 sube a 35x más rápido
pcnt9	2298	633	638	826	pcnt9	-	-	-	21,50	SSE4 sólo 30x por leer 32b
pcnt10	1138	436	423	415	pcnt10	-	40,75	42,01	42,85	SSE4 128b sube a 44x

bucles for/while



sumas en árbol



repertorio multimedia

