

Nombre: Jia Long Ji Qiu

Grupo: 23

Nombre: Jiabo Wang

## Hoja de respuestas de la práctica

1. Instrucciones dinámicas del código de ejemplo `Simple.c`:

Sin optimizar: 15,191,215

Optimizado: 191,489

Explicación de los resultados:

Como era de esperarse, tras haber compilado `Simple.c` de una manera optimizada (con `-O2`), el número de instrucciones dinámicas se ha reducido. Esto se debe a que durante el proceso de compilación se han ejecutado algoritmos de compilación que han permitido reducir código.

2. Rellena la siguiente tabla sobre el programa `Poker.c`:

	Tiempo	Ciclos	Instrucciones	MIPS	CPI	Frec.	Speedup
-O0	357.54 ms	$1,22 \cdot 10^9$	2,500,607,866	6994	0.488	3.41GHz	1x
-O2	95.45 ms	$325 \cdot 10^6$	724,337,156	7589	0.449	3.40GHz	3.75x
-O2 + "inlining"	76 ms	$259 \cdot 10^6$	646,372,490	8505	0.4	3.41GHz	4.7x

3. Rellena la siguiente tabla sobre la rutina `PierdeTiempo` (tened cuidado de no incluir el `printf` en los ciclos de la rutina) del programa `Poker.c`:

	Ciclos Programa	Ciclos rutina	% de Ciclos	Speedup teórico máximo
-O0	$1,22 \cdot 10^9$	$438 \cdot 10^6$	35,9%	1.56x
-O2	$325 \cdot 10^6$	24	$7.38 \cdot 10^{-6}\%$	1x

Las variaciones entre las dos filas del apartado anterior se deben a:

En el caso de `-O0` la rutina `PierdeTiempo()` ocupa un porcentaje mayor que en el caso de `-O2`. Por lo que el Speedup tras reducir el tiempo de dicha rutina es más notable.