TRABAJO EN LABORATORIO

Para probar el funcionamiento de la memoria cache que has diseñado en el informe previo (estructura MemCacheDirect y función callCache) la llamaremos desde un programa principal main que hace loads y store de datos. Este programa principal lee los datos de un vector X de 24 posiciones y los suma 1 a 1 a los 24 datos de otro vector Y de también 24 posiciones. El resultado de la suma de estos dos vectores será el vector Z.
En principio el programa sería:

```
main(){
int i;
for(i=0; i<24;i++)
        {
        Z[i]=X[i]+Y[i];
        }
}
```

Pero nosotros introduciremos llamadas a la función callCache en cada acceso a memoria y además contaremos el número de hits(aciertos) y el número de misses (fallos). Sería algo así:

```
main(){
int i, contador_hit, contador_miss;
for(i=0; i<24;i++)
        {
        Z[i]=X[i]+Y[i];
        if(callCache(@X[i])==1) contador_hit++;
        else contador_miss++;
        if(callCache(@Y[i])==1) contador_hit++;
        else contador_miss++;
        if(callCache(@Z[i])==1) contador_hit++;
        else contador_miss++;
        }
}
```

1) Abre el archivo practica2.s. En este archivo ya están en .data puestos los valores de los vectores X e Y. Copia o completa la función callCache que has definido en el apartado g) del informe previo, así como la estructura de MemCacheDirect.
2) Traduce a ensamblador MIPS el programa main e impleméntalo en el archivo pratica2.s. Cuando veas que funciona sin dar errores llama al profesor. Copia en el archivo y aquí el programa main que has creado.

3) Simularemos paso a paso contestando a las siguientes preguntas:
   a) ¿En qué dirección de memoria empieza la estructura MemCacheDirect?
   b) ¿Cuántos bytes ocupa la estructura MemCacheDirect?

   c) ¿En qué direcciones de memoria están los vectores X, Y, Z?

d) ¿Cuándo se produce el primer acierto de cache?

e) En la 10º iteración del bucle, a qué direcciones de los vectores X, Y y Z se acceder?

En esta misma 10ª iteración, cuando llamamos a callCache(@X[9]), en qué posición de memoria está MemCacheDirect[line_cache(@X[9]).VALID y MemCacheDirect[line_cache(@X[9]).TAG?

¿Qué dirección de línea de cache es para line_cache(@X[9])?

¿Se produce fallo o acierto al acceder a line_cache(@X[9])?

f) ¿Cuántos fallos y aciertos en total se producen? ¿Cuántos de X, cuántos de Y y cuántos de Z?

g) Puedes observar algún patrón de comportamiento en estos accesos a memoria.

4) Suponemos que los aciertos en cache tienen una duración de 1 ciclo en caso de lectura y los fallos tanto en lectura como en escritura son de 10 ciclos. En el caso de escritura aunque acierte tarda 10 ciclos pues asumimos que si acierta escribe a la vez en memoria cache y memoria principal. Teniendo en cuenta los datos que has obtenido de la simulación: aciertos y fallos X (lecturas), y fallos Y (lecturas) y aciertos y fallos Z (escrituras). ¿Cuál sería el tiempo en realizar todos estos accesos?