

Trabajo práctico 1: Conjunto de instrucciones MIPS

1 Introducción

Este trabajo práctico tiene como objetivo familiarizarse con el conjunto de instrucciones MIPS 32, así como identificar posibilidades de mejora sobre la solución obtenida.

En la clase del jueves 4/4 presentamos las herramientas necesarias para poder compilar y ejecutar programas escritos en assembly MIPS 32 y el ABI requerido para poder ejecutar llamadas a función desde código C utilizando el compilador GCC

2 Descripción

El trabajo práctico consiste en implementar un problema denominado **"La hormiga artista"**. El mismo plantea un recorrido sobre un espacio de dos dimensiones, donde el mismo está definido por reglas predefinidas.

Se dispone de un area bidimensional dividida en celdas cuyo tamaño es configurable, donde cada celda está pintada de un color ¹. En la posición central se encuentra una hormiga, orientada hacia el norte², que dispone de una paleta de colores. Al encontrarse sobre una celda, la hormiga cambia su orientación rotando hacia la izquierda (L) o derecha (R) según el color sobre el que se encuentre, pinta la celda utilizando el siguiente color de la paleta ³ y se mueve a la celda que tiene adelante. Esta secuencia de acciones se repite un número de veces predeterminado. Finalmente se escribe el estado de la grilla por la salida estandar, en formato PPM.

La paleta de colores contiene una combinación de los siguientes colores: Rojo (R), Azul (B), Verde (G), Amarillo (Y), Blanco (W) y Negro (N).

¹El color inicial de todo el espacio es el primero de la paleta especificada

²En el formato de salida, hacia el encabezado del archivo

³El primer color a utilizar es el segundo color de la paleta

3 El programa

El programa debe tomar las siguientes opciones

- Las dimensiones de la grilla
- La paleta de colores
- El conjunto de reglas para realizar las rotaciones
- La cantidad de movimientos a realizar

3.1 Implementación

Como punto de partida se provee un esqueleto escrito en C en el cual se encuentran resueltos aspectos como la evaluación de parámetros, algunas estructuras y la impresión final en formato PPM. Sin embargo, el movimiento de la hormiga a través del espacio bidimensional debe programarse y tiene la siguiente interfaz:

```
void* paint(void *ant, void *grid, void *palette, void *rules, uint32_t times);
```

Asimismo, la implementación de la paleta de colores y el conjunto de reglas debe ser realizado como parte de la entrega. Existen dos funciones encargadas de interpretar dichos parámetros - `make_palette(char *)`, `make_grid(char *)` - que en su versión *default* detienen la ejecución del programa.

Dentro del archivo `ant_constants.h` contiene constantes predeterminadas para las orientaciones, rotaciones y colores.

Como parte de la entrega se pide escribir dos versiones de este programa. La primera, escrita íntegramente en C, completando el esqueleto. La segunda versión, implementando la función `paint` en assembly MIPS.

3.1.1 Código MIPS

Se debe disponer de una implementación de la función `paint` en código *assembly* MIPS, que respete el ABI de la cátedra. Cualquier función llamada desde dicha función debe estar implementada en *assembly*. Si bien es posible llamar a funciones C desde código assembly, es preferible limitar su uso a funciones de depuración.

Con el fin de obtener un programa *correcto*, se recomienda escribir una versión inicial que no contemple optimizaciones prematuras. Por ejemplo, utilizando el *stack* para variables locales y realizando una traducción de código C a *assembly*. Posteriores versiones pueden implementar mejoras como tablas de saltos, variables locales en registros o variaciones al ABI.

4 Ejemplos

```
./tp1 --help
./tp1 -g <dimensions> -p <colors> -r <rules> -t <n>
-g --grid: wxh
-p --palette: Combination of R|G|B|Y|N|W
-r --rules: Combination of L|R
-n --times: Iterations
-h --help: Print this message and exit
-v --verbose: Version number

$ tp1 --grid 3x3 --palette "R|G|B" --rules "L|R|L" --times 3
P3
3 3
255

255 0 0 255 0 0 255 0 0
0 0 255 0 255 0 255 0 0
255 0 0 255 0 0 255 0 0

$ tp1 --grid 3x3 --palette Y|N|B --rules L|R|L --times 2
P3
3 3
255

255 255 0 255 255 0 255 255 0
0 0 255 0 0 0 255 255 0
255 255 0 255 255 0 255 255 0

$ tp1 --grid 2x2 --palette N|W --rules R|L --times 1
P3
2 2
255

0 0 0 0 0 0
0 0 0 255 255 255
```

5 Extras

- Mediciones de tiempo
- Impacto sobre el *cache*

6 Recursos

- Formato PPM: <http://netpbm.sourceforge.net/doc/ppm.html>

7 Condiciones de entrega

Las fechas de entrega para el trabajo práctico son el Jueves 25/4 y el jueves 9/5. La entrega debe incluir un informe describiendo la resolución del trabajo práctico, que incluya:

- Carátula especificando los datos y contacto de los integrantes del grupo (dirección de correo electrónico, *handle* de slack)
- Instrucciones necesarias para compilar y ejecutar el programa
- Decisiones relevantes sobre la implementación y resolución
- Conclusiones con fundamentos reales
- Diagramas ilustrando la estructura del *stack* de cada función relevante
- Casos de prueba documentados
- Código fuente
- Este enunciado