

Trabajo práctico 1A: Conjunto de instrucciones MIPS

1. Objetivo

El objetivo de este trabajo es proveer una versión del programa en la que las funciones `new_orientation` y `move_forward` estén codificadas en assembly MIPS32, y comparar su desempeño con el de esas mismas funciones compiladas con diversos grados de optimización.

2. Introducción

Este trabajo práctico toma como precedente el Trabajo Práctico 0, que evaluó el desempeño de dos implementaciones posibles de un problema denominado "La hormiga artista". Además se ejercitó la instalación de un ambiente capaz de ejecutar programas para distintas variantes de la arquitectura MIPS.

En la clase del 12/9 se presentó la convención de llamadas a funciones a ser utilizada por la práctica. Partiendo desde este punto, se busca extender el trabajo práctico implementando una parte acotada de la funcionalidad en assembly MIPS32.

Al finalizar el trabajo práctico se deben presentar conclusiones relevantes sobre la implementación realizada incluyendo ventajas y desventajas de la implementación en este lenguaje.

3. Implementación

En el trabajo práctico anterior, existían dos versiones de las siguientes funciones, implementadas en C. En este trabajo las mismas deben estar codificadas en assembly MIPS32, respetando el ABI de la cátedra.

```
orientation_t
new_orientation(orientation_t orientation, rotation_t rule);

ant_t*
move_forward(ant_t* ant, uint32_t width, uint32_t height);
```

3.1. Ejemplos

Listamos las opciones utilizando el comando `--help`

```
./tp1_if --help
./tp1_if -g <grid_spec> -p <colour_spec> -r <rule_spec> -t <n>
-g --grid: wxh
-p --palette: Combination of RGBYNW
-r --rules: Combination of LR
-t --times: Iterations. If negative, its complement will be used.
-o --outfile: output file. Defaults to stdout.
-h --help: Print this message and exit
-v --verbose: Version number
```

Compile with `-DSANITY_CHECK` to enable runtime checks

Compile with `-DUSE_TABLES` to execute ant operations in separate functions

Compile with `-DUSE_COL_MAJOR` to traverse the grid in column-major order

Medimos el tiempo en ejecutar diez mil operaciones en la menor grilla posible, y repetimos escalando la cantidad de operaciones

```
time -p ./tp1_if -g 1x1 -p RGBW -r LLLL -t ((10 * 1000)) > /dev/null
```

```
time -p ./tp1_if -g 1x1 -p RGBW -r LLLL -t $((100 * 1000)) > /dev/null
```

```
time -p ./tp1_if -g 1x1 -p RGBW -r LLLL -t $((1000 * 1000)) > /dev/null
```

Repetimos, con una grilla significativamente mas grande

```
time ./tp1_if -g 1024x1024 -p RGBW -r LLLL -t $((10 * 1000)) > /dev/null
```

```
time ./tp1_if -g 1024x1024 -p RGBW -r LLLL -t $((100 * 1000)) > /dev/null
```

```
time ./tp1_if -g 1024x1024 -p RGBW -r LLLL -t $((1000 * 1000)) > /dev/null
```

4. Análisis de desempeño

Compilar la versión en C del programa con los grados de optimización `-O0`, `-O1`, `-O2` y `-O3`, conservando los distintos ejecutables. Graficar el tiempo utilizado por la versión que contiene código assembly y la versión en C compilada con las distintas optimizaciones, para una matriz de $1024 * 1024$ y con 10^n iteraciones, para n entre 4 y 9.

5. Condiciones de entrega

El trabajo práctico vence el 03/10/2019. Debe contener:

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

6. Recursos

- Hormiga de Langton: https://es.wikipedia.org/wiki/Hormiga_de_Langton
- Formato PPM: <http://netpbm.sourceforge.net/doc/ppm.html>
- Imagemagick <https://imagemagick.org/index.php>
- MIPS32 Instruction Reference <http://www.mrc.uidaho.edu/mrc/people/jff/digital/MIPSir.html>
- Convención de llamadas a funciones, disponibles en el grupo Yahoo de la cátedra (Files >Material >Assembly MIPS >func_call_conv.pdf)
- GDB Debugger <https://www.gnu.org/software/gdb/>
- Objdump <https://linux.die.net/man/1/objdump>