

66:20 Organización de Computadoras

Trabajo práctico 1: Conjunto de instrucciones MIPS

1. Objetivos

Familiarizarse con la programación en assembly y el concepto de ABI, implementando una versión simplificada del programa `sha1`[1], descrito en la sección 5.

2. Alcance

Este trabajo práctico es de elaboración grupal, evaluación individual, y de carácter obligatorio para todos alumnos del curso.

3. Requisitos

El trabajo deberá ser entregado personalmente, en la fecha estipulada, con una carátula que contenga los datos completos de todos los integrantes.

Además, es necesario que el trabajo práctico incluya (entre otras cosas, ver sección 7), la presentación de los resultados obtenidos, explicando, cuando corresponda, con fundamentos reales, las causas o razones de cada resultado obtenido.

El informe deberá respetar el modelo de referencia que se encuentra en el grupo¹, y se valorarán aquellos escritos usando la herramienta \TeX / \LaTeX .

4. Recursos

Usaremos el programa GXemul[2] para simular el entorno de desarrollo que utilizaremos en este y otros trabajos prácticos, una máquina MIPS corriendo una versión reciente del sistema operativo NetBSD[3].

5. Descripción

El programa a escribir es una versión minimalista del programa `sha1`, disponible en el entorno de desarrollo. El mismo genera un checksum a partir de los

¹<http://groups.yahoo.com/group/orga6620>

contenidos de un archivo, utilizando el algoritmo SHA-1[1]. El programa puede recibir 0,1 o más archivos.

En caso de no recibir un nombre de archivo, lee de `stdin`. Al recibir uno o más archivos, informa el nombre de cada archivo junto con su checksum. Todos los errores deben escribirse en `stderr`.

En la clase del 9/4 se explicó el concepto de ABI. La misma debe respetarse al realizar este trabajo práctico.

6. Implementación

6.1. Programa

Si bien la interpretación de argumentos del programa y la lectura de archivos puede realizarse en C, la función que calcula el checksum debe ser implementada en Assembly MIPS. Ésta debe tener el siguiente prototipo:

```
int sha1(char *result, char *bytes, unsigned long length)
```

Sus argumentos son un puntero a `char` que tenga espacio suficiente para escribir el resultado, un puntero a `char` que apunta al comienzo de los contenidos del archivo, y un `long` que representa el largo del archivo. La función debe escribir el checksum como caracteres hexadecimales en `result`, y devolver 0 como resultado si no hubo errores, o 1 si los hubiere. El programa debe imprimir el checksum por `stdout`

6.2. Ejemplos

Primero, usamos la opción `-h` para ver el mensaje de ayuda:

```
$ tp1 -h
Usage:
  tp1 -h
  tp1 -V
  tp1 [file...]
Options:
  -V, --version      Print version and quit.
  -h, --help         Print this information and quit.
```

```
Examples:
  tp1 foo
  echo "hello" | tp1
```

Ejemplos de uso:

```
$ echo "hello" | ./tp1
f572d396fae9206628714fb2ce00f72e94f2258f

$ cat in.1
Don't panic
$ cat in.2
So long, and thanks for all the fish
$ sha1 in.1 in.2
SHA1 (in.1) = 09ae937402aac2b22427e279c539f0a3f1ebedc4
SHA1 (in.2) = 47ae91276b0f80a23b08bbced3479184761af706
```

6.3. Portabilidad

Como es usual, es necesario que la implementación desarrollada provea un grado mínimo de portabilidad. Para satisfacer esto, el programa deberá funcionar al menos en NetBSD/pmax (usando el simulador GXemul[2]) y la versión de Linux (Knoppix, RedHat, Debian, Ubuntu) usada para correr el simulador, Linux/i386.

7. Informe

El informe deberá incluir:

- Documentación relevante al diseño e implementación del programa;
- Comando(s) para compilar el programa;
- Las corridas de prueba, con los comentarios pertinentes;
- Diagramas indicando el stack de cada función implementada en assembly.
- El código fuente.
- Este enunciado.
- Un CD conteniendo todo el material digital. Por favor no pegarle etiquetas.

8. Fechas

- Primera entrega: Jueves 30 de Abril.
- Revisión: Jueves 7 de Mayo.
- Vencimiento: Jueves 14 de Mayo.

Referencias

- [1] SHA-1 (Wikipedia). <http://en.wikipedia.org/wiki/SHA-1>.
- [2] GXemul, <http://gavare.se/gxemul/>.
- [3] The NetBSD project, <http://www.netbsd.org/>.