

# Convenciones a ser usadas en el Taller

## Uso de la pila

### Pila

La pila crece en el sentido de direcciones altas a direcciones bajas.

### Marco de ejecución

El marco de ejecución de una rutina se refiere al conjunto de localidades de memoria (dentro de la pila) que esa rutina administra, por lo tanto es esa rutina la que empiló los valores contenidos en esas localidades, asimismo, esa misma rutina es la responsable de desempilarlos.

### Frame pointer (\$fp)

apunta a la primera dirección del marco de la rutina en ejecución.

### Stack pointer (\$sp)

apunta a la primera posición libre de la pila (la posición siguiente al final del marco de la rutina en ejecución).

## Convención con responsabilidad compartida

- El procedimiento llamador (caller) en un momento dado será el que invoca a otro procedimiento. El llamador es responsable de los registros.

\$a0, \$a1, \$a2, \$a3, \$v0, \$v1, \$t0, ..., \$t9

- El procedimiento llamado (callee) es la rutina que fue invocada por otra. El llamado será responsable de los registros.

\$s0, ... , \$s7, \$fp, \$ra

## La convención que vamos a utilizar debe ser tomada en cuenta en cuatro puntos:

- I. Antes de que el "llamador" invoque al "llamado"
- II. Justo antes de que el "llamado" comience su ejecución
- III. Inmediatamente antes de que el "llamado" retorne al "llamador"
- IV. Después de que el "llamador" invoque al "llamado"

## I. Antes de que el "llamador" invoque al "llamado"

1. Salva los registros que son responsabilidad del "llamador" sí el contenido de éstos es necesario después de la llamada (\$a0,...\$a3,\$t0,...\$t9).
2. Lleva a cabo el pasaje de argumentos.
  1. Los primeros cuatro argumentos son pasados en los registros (\$a0,...,\$a3 y éstos son usados en ese orden, por ejemplo: sí una rutina requiere 2 parámetros usara \$a0 y \$a1).
  2. el resto de los argumentos (del quinto en adelante) se pasan usando la pila. Los argumentos son empilados quedando el quinto argumento en el tope de la pila, es decir, es el último que se empila (recuerde que según la definición de pila, esta crece hacia las direcciones bajas).
3. Ejecuta la instrucción jal.

## II. Justo antes de que el "llamado" comience su ejecución (Prólogo)

1. Almacenar en la pila los registros que son responsabilidad del "llamado" (\$fp, \$ra, \$s0, ..., \$s7) y que son utilizados en la rutina. Siguiendo las siguientes consideraciones:
  - El registro \$fp será almacenado en la pila solamente sí el "llamado" empilará algún registro.
  - El registro \$ra será almacenado en la pila solamente sí el "llamado" invoca a su vez a otro procedimiento.
  - Los otros registros que son responsabilidad del llamado se salvan sí su contenido va a ser alterado.

**Nota:** estos registros no se usan para pasar valores globales, así que no se puede asumir que tengan un valor determinado.

Mueve el \$fp a la primera dirección del marco o bloque que le corresponde a la rutina en ejecución (al comienzo del marco del llamado).

## Inmediatamente antes de que el "llamado" retorne al "llamador" (Epílogo)

Si el "llamado" es una rutina que retorna algún valor, éstos serán devueltos siguiendo el siguiente esquema:

- sí retorna un solo valor, éste debe ser colocado en \$v0.

- sí retorna dos valores, el primero debe ser colocado en \$v0 y el segundo en \$v1.

Restaura todos los registros responsabilidad del "llamado" que fueron almacenados en la pila. La pila debe quedar en el mismo estado en que fue recibido.

### **Después de que el "llamador" invoque al "llamado"**

Restaurar de la pila los registros responsabilidad del "llamador" que fueron colocados en la pila.

En cuanto a los argumentos que fueron pasados por la pila serán desempilados por el "llamador" ya que pertenecen a su marco de ejecución, es decir, quien los colocó en la pila.