

Nombre: Pol Casauberta Gil
Nombre: Victor Valeriev

Grupo: 33

Hoja de respuesta al Estudio Previo

1. Explica en detalle qué hacen las siguientes instrucciones. Si están sintácticamente mal escritas, indica el motivo:

`movl $1, %eax`

Guarda en el registro `eax` el literal 1

`movl 1, %eax`

Al inmediato le falta el símbolo `$`

`movl $1, eax`

Para referirse al registro `eax` hay que añadir el `"%"`

`movl 1, eax`

Contiene ambos errores anteriores

2. Dada la siguiente secuencia de instrucciones:

`cmpl %eax, %ebx`

`jge fin`

¿en qué condiciones se efectúa el salto?

Saltaría en caso que `ebx` sea mayor o igual a `eax`

3. Dada la siguiente secuencia de instrucciones:

`cmpl %ebx, %eax`

`jge fin`

¿en qué condiciones se efectúa el salto?

Saltaría en caso que `eax` sea mayor o igual a `ebx`

4. La traducción a código ensamblador del código C es:

`movl a, %ebx
movl b, %ecx
movl $0, %edx`

`for:
cmpl $300, %edx
jge fm
cmpl %ebx, %ecx
jne endfor`

`if: movl %ecx, %esi
subl %ebx, %esi
shra 2, %esi
addl %esi, %ebx
endif:
incl %edx
jmp for
endfor:`

$$e_i = b$$
$$e_i = b - a$$
$$e_i = \frac{b-a}{2}$$
$$e_i = \frac{b+a}{2}$$

5. La traducción a código C del código ensamblador es:

```
while (i > 0) {  
    if (a != b) {  
        b = b - a;  
        a = b;  
        a = (a * 2) + b;  
    }  
    i = i - 1;  
}
```

6. Explica cuál es la diferencia entre un breakpoint y un watchpoint.

Un breakpoint indica una línea de código donde quieres que la ejecución se detenga. Un watchpoint indica un dato a observar y cuando el dato cambia, el programa se detiene.

7. Explica cómo se puede hacer en ddd que un breakpoint sólo se pare cuando se ha ejecutado 15 veces.

Añades un breakpoint en la línea deseada y en las propiedades le pones un ignore count = 15 y luego "apply".

8. Explica cómo se puede hacer en ddd para modificar el contenido de una dirección de memoria.

Usando el comando set *((int*)"address") = "value"