

Nombre: Jia Long Ji Qiu

Grupo: 23

Nombre: Jiabo Wang

Hoja de respuesta al Estudio Previo

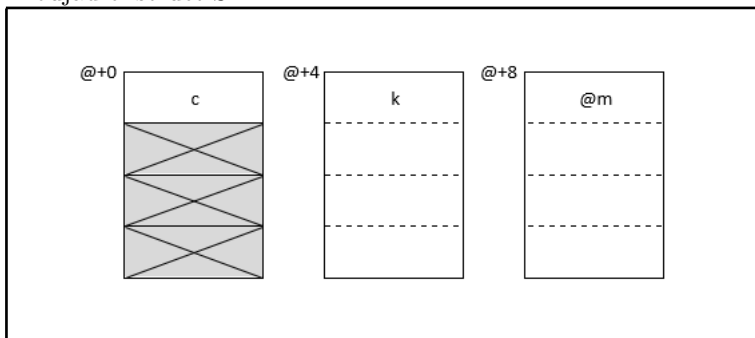
1. Explicad qué significan y para qué sirven los parámetros “argc” y “argv” del main.

“argc” es el “argument count”, guarda el número de parámetros que se pasa a la rutina main de un programa. “argv” son los “argument values”, y es un puntero que hace referencia a un vector que guarda los parámetros que se han pasado en dicha rutina.

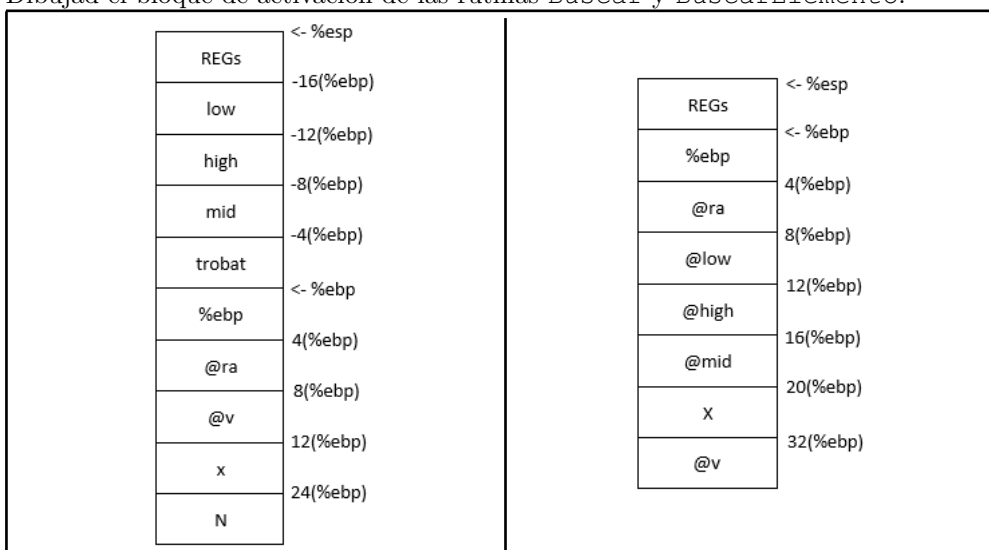
2. Explicad para qué sirve la función atoi

atoi() es una función que sirve para hacer una conversión de una variable de tipo string a otra de tipo entero. Por ejemplo, atoi("1") devuelve el valor entero 1.

3. Dibujad el struct S1:



4. Dibujad el bloque de activación de las rutinas Buscar y BuscarElemento.



5. Dibujad el bloque de activación de la rutina main.

	<- %esp
REGs	-412(%ebp)
a	-408(%ebp)
b	-404(%ebp)
n	-400(%ebp)
trobat	-396(%ebp)
X	-384(%ebp)
Y	-372(%ebp)
Z	-360(%ebp)
v	<- %ebp
%ebp	4(%ebp)
@ra	8(%ebp)
argc	12(%ebp)
@argv	

6. El bloque de activación y el código de SencillaSub en ensamblador del x86 es:

	<- %esp	
REGs	-4(%ebp)	
i	<- %ebp	
%ebp	4(%ebp)	
@ra	8(%ebp)	
a	20(%ebp)	
b		


```

pushl %ebp
movl %esp, %ebp
subl 4, %esp
movb 20(%ebp), %eax
if:  cmpb 8(%ebp), %eax
     jne else
     movl $0, -4(%ebp)
     jmp endif
else: movl 16(%ebp), %ecx
     movl (%ecx), %ecx
     movl %ecx, -4(%ebp)
endif: movl -4(%ebp), %eax
     addl 12(%ebp), %eax
     movl %ebp, %esp
     popl %ebp
     ret

```

7. El bloque de activación y el código de LlamaSencilla en ensamblador del x86 es:

REGs	<- %esp
%ebp	<- %ebp
@ra	4(%ebp)
x	8(%ebp)
y	20(%ebp)

```
pushl %ebp
movl %esp, %ebp
pushl 8(%ebp)
pushl 16(%ebp)
pushl 12(%ebp)
pushl 8(%ebp)
call SencillaSub
movl 16(%ebp), %ecx
movl %eax, (%ecx)
movl 12(%ebp), %eax
movl %ebp, %esp
popl %ebp
ret
```