

Nº A92846

Nome:

Carlos Riquel Passos Ferreira

Turma: Turno 1

**Resolução dos exercícios (deve ser redigido manualmente)****1. Código em assembly**

Escreva aqui o código otimizado em *assembly* (tal como está no ficheiro `while_loop.s`) devidamente anotado, i.e., com comentários à frente de cada instrução, comentários esses que

(i) expliquem o que está a acontecer se for a fase de arranque ou término duma função e

(ii) mostrem que parte do código C essa instrução em *assembly* está a executar.

De seguida, analise o código em *assembly* e preencha a tabela.

(escrito na próxima página)

[?] ↓

Variável	Registo	Atribuição inicial
x	%ebx	x
y	%eax	y
n	%edx	n

Escreva aqui o código C de um programa simples (main) que usa a função while loop:

```

int main ()
{
    while_loop (1, 2, 3)
}

```

→ • indica que a função "main()" pode retornar um valor do tipo "integer"

• se usasse "void", a função não ia retornar nenhum valor.

1.

while-loop:

```

pushl %ebp
movl %esp, %ebp
movl 16(%ebp), %edx
testl %edx, %edx
pushl %ebx
movl 12(%ebp), %eax
movl 8(%ebp), %ebx
jle .L3
movl %edx, %ecx
sall $4, %ecx
compl %ecx, %eax
jge .L3
p2align 2, 3

```

# Arranque

#  $edx = n$

#  $n \& n$  (?)

# o valor de %ebx passa a estar no topo da stack

#  $eax = y$

#  $ebx = x$   $SF \neq OF \mid ZF = 1$

# if  $n \leq 0$  ( $SF \wedge OF \mid ZF$  ???), goto ".L3"

#  $ecx = n$

#  $ecx = n * 2^4 = n * 16$

# comparar  $y : (n * 16)$

# if  $y \geq n * 16$ , goto ".L3"

# (???)

.L6:

```

addl %edx, %ebx
imull %edx, %eax
decl %edx
subl $16, %ecx
testl %edx, %edx
jle .L3
compl %ecx, %eax
jl .L6

```

#  $x = n + x$

#  $y = y * n$

#  $n = n - 1$  ( $n--$ )

#  $ecx = ecx - 16$  (?)

#  $n \& n$

# if  $y \leq 0$ , goto ".L3"

# comparar  $y : (n * 16)$

# if  $y < n * 16$ , goto ".L6"

.L3:

```

movl %ebx, %eax
popl %ebx
leave
ret

```

# devolve o valor de  $x$

} término

função "test" define a Flag a 0 quando o resultado da operação "&" (AND) for 0.  
(?)

Após completar o ficheiro “while\_loop.c” com o programa main, tal como mostra a próxima figura

```
int while_loop(int x, int y, int n)
{
    while ((n>0) && (y<16*n))
    {
        x += n;
        y *= n;
        n--;
    }
    return x;
}

int main()
{
    while_loop(4,2,3);
}
~
~
~
~
~
```

Surgiram uma série de erros após mandar executar o seguinte código:

```
[03a92846@sc TPC6]$ gcc -Wall -O2 -g while_loop while_loop.c
```

Os erros que falo são os seguintes:

```
gcc: while_loop: No such file or directory
while_loop.c: In function `main':
while_loop.c:19: warning: control reaches end of non-void function
```

Até à hora limite de entrega, não consegui resolver o problema, pelo que não consegui completar o TPC6.