

Universidad Nacional de San Antonio Abad del Cusco



CURSO: Organización y Arquitectura del Computador

Informe: INVOCACION DE FUNCIONES ESCRITAS EN ENSAMBLADOR DESDE C

Alumno Luis Abelardo Jara Hillpa

Código: 160544

Docente: Edwin Carrasco Poblete

Octubre 2020

1. Escriba un programa que muestre los N primeros números primos. El módulo para determinar si un número es primo debe escribirse en ensamblador. N es ingresado por el usuario.

El prototipo del módulo ensamblador debe ser:

```
bool esPrimo(int N);
```

bool puede ser:

1 – Verdadero

0 – Falso

CODIGO FUENTE:

Ensamblador:

```
;nombre: primoc.asm
```

```
;propósito: este programa evalua si un nuermo es primo o no
```

```
;autor: Luis Abelardo Jara Hillpa
```

```
;f. creacion: 13/10/2020
```

```
;f. modificación: -----
```

```
;compilación: nasm -f elf primoc.asm
```

```
section .bss
```

```
    numero    resd    1
```

```
section .text
```

```
    global primo
```

```
primo:
```

```
    ;crear marco de pila
```

```
    push ebp
```

```
    mov ebp, esp
```

```
    mov eax, [ebp+8]
```

```
    mov [numero], eax
```

```
    mov ebx, eax
```

```
    mov ecx, 0
```

p1:

```
xor edx,edx
div ebx
push edx
mul ebx
pop edx
add eax, edx
sub ebx, 1
cmp ebx, 0
je evaluar
cmp edx, 0
je contar
jmp p1
```

contar:

```
add ecx, 1
jmp p1
```

evaluar:

```
cmp ecx, 1
je esprimo
;retorna 0 si no es primo
mov eax, 0
jmp fin
```

esprimo:

```
;retorna 1 si es primo
mov eax, 1
```

fin:

```
pop ebp
ret
```

C:

//nombre: esprimo.c

//propósito: muestra los primeros N números primos utilizando una func en ensamblador

//autor: Luis Abelardo Jara Hillpa

//f. creacion: 13/10/2020

//f. modificación: -----

//compilar: gcc -o primos esprimo.c primoc.o

//ejecución: ./primos

#include <stdio.h>

#include <stdbool.h>

extern bool primo(int);

main()

{

 //variables

 int n;

 int contador=0;

 int contador2=1;

 //indicar que hace el programa

 printf("ESTE PROGRAMA MUETRA LOS N PRIMEROS NUMEROS PRIMOS\n");

 //leer datos

 printf("ingrese N:");

 scanf("%d", &n);

 //evaluar

 while (contador<n)

 {

 if (primo(contador2))

 {

 printf("%d\n", contador2);

 contador++;

 }

```

        contador2++;
    }
}

```

EJECUCIÓN:

```

guest@porteus:~$ nasm -f elf primoc.asm
guest@porteus:~$ gcc -o primos esprimo.c primoc.o
guest@porteus:~$ ./primos
ESTE MUESTRA LOS N PRIMEROS NUMEROS PRIMOS
ingrese N :12
2
3
5
7
11
13
17
19
23
29
31
37
guest@porteus:~$ █

```

- Si la suma de los divisores de un número M (excepto M) es igual a N y la suma de los divisores de N (excepto N) es igual a M, se dice que estos son números amigos; por ejemplo 220 y 284 son números amigos. Escriba un programa que determine todos los números amigos desde 1 hasta B. El módulo que determine si dos números son amigos debe escribirse en ensamblador. B es proporcionado por el usuario.

El prototipo del módulo ensamblador debe ser:

```
bool sonAmigos(int N, int M);
```

bool puede ser:

1 – Verdadero

0 – Falso

CODIGO FUENTE:

Ensamblador:

;nombre: amigosc.asm

;proposito: este programa evalua si dos numeros son amigos

;autor: Luis Abelardo Jara Hillpa

;f.creacion: 13/10/2020

;f.modf: -----

;compilacion: nasm -f elf amigosc.asm

section .bss

num1 resd 1

num2 resd 1

section .text

global amigos

amigos:

;crear marco de pila

push ebp

mov ebp, esp

mov eax, [ebp+8]

mov [num1], eax

mov ebx, [ebp+12]

mov [num2], ebx

push eax

push ebx

mov ebx, eax

xor ecx, ecx

dec ebx

sd1:

cmp ebx, 0

je s1

xor edx, edx

div ebx

push edx

cmp edx, 0

je sum1

l1:

xor edx, edx

mul ebx

pop edx

add eax, edx

dec ebx

jmp sd1

sum1:

add ecx, ebx

jmp l1

s1:

pop eax

cmp ecx, eax

je trance

;retornar 0 si los numeros no son amigos

mov eax, 0

pop edx

jmp salir

trance:

mov ebx, eax

dec ebx

xor ecx, ecx

sd2:

cmp ebx, 0

je s2

xor edx, edx

div ebx

push edx

cmp edx, 0

je sum2

l2:

```
xor edx, edx
```

```
mul ebx
```

```
pop edx
```

```
add eax, edx
```

```
dec ebx
```

```
jmp sd2
```

```
sum2:
```

```
add ecx, ebx
```

```
jmp l2
```

```
s2:
```

```
pop eax
```

```
cmp ecx, eax
```

```
je Esamigo
```

```
;retornar 0 si los números no son amigos
```

```
mov eax, 0
```

```
jmp salir
```

```
Esamigo:
```

```
;retornar 1 si los números son amigos
```

```
mov eax, 1
```

```
salir:
```

```
pop ebp
```

```
ret
```

C:

```
//nombre: sonamigos.c
```

```
//propósito: muestra los números amigos entre 1 y un numero ingresado por el usuario una func en  
ensamblador
```

```
//autor: Luis Abelardo Jara Hillpa
```

```
//f. creacion: 13/10/2020
```

```
//f. modificación: -----
```

```
//compilar: gcc -o amigos sonamigos.c amigosc.o
```


//ejecución: ./amigos

#include <stdio.h>

#include <stdbool.h>

extern bool amigos(int, int);

main()

{

 //variables

 int limite, m, n;

 //indicar que hace el programa

 printf("ESTE PROGRAMA MUESTRA LOS NUMEROS AMIGOS ENTRE 1 HASTA UN NUMERO
INGRESADO POR EL USUARIO");

 //leer datos

 printf("ingrese el limite: ");

 scanf("%d", &limite);

 //Evaluar

 for (m=2;m<limite; m++)

 {

 for (n=2;n<limite; n++)

 {

 if (m==n)

 {

 if (amigos(m,n)==true)

 {

 printf("%d y %d\n",m ,n);

 }

 }

 }

 }

}

EJECUCIÓN:

```
guest@porteus:~$ nasm -f elf amigosc.asm
guest@porteus:~$ gcc -o amigos sonamigos.c amigosc.o
guest@porteus:~$ ./amigos
ESTE PROGRAMA MUESTRA LOS NUMEROS AMIGOS ENTRE 1 HASTA UN NUERMO INGRESADO POR EL USUARIO
Ingrese el limite: 2000
220 y 284
284 y 220
1184 y 1210
1210 y 1184
guest@porteus:~$
```