

Ejercicios del Capítulo 5

Práctica 5

M. A. Noriega Vargas, C. D. Ruiz Guerrero,
A. Ruiz Medina, A. Rodríguez Buenrostro

1 Objetivo

-Realizar una serie de programas con funciones recursivas.

2 Introducción

La recursividad es una técnica que se utiliza en programación que le permite al programador que un bloque de instrucciones se ejecute un cierto número de veces que este ha indicado. Se dice que una función es recursiva cuando esta se llama así misma. Un algoritmo recursivo es un algoritmo que expresa la solución de un problema en términos de una llamada a sí mismo. La llamada a sí mismo se conoce como llamada recursiva o recurrente.

3 Ejercicios

3.1 Factorial

```
#include <stdio.h>

long int fac(int n);

int main(){
    int n;

    printf("Ingrese un entero positivo para calcular su factorial: ");
    scanf("%d", &n);
    printf("\nEl factorial de %d es %ld\n", n, fac(n));
    return 0;
}
```

```

long int fac(int n){
    if (n >= 1)
        return n*fac(n-1);
    else
        return 1;
}

```

```

laptop01@laptop01-Aspire-E5-511P: ~/Practicas/Capitulo5
laptop01@laptop01-Aspire-E5-511P:~$ cd Practicas/Capitulo5/
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ gcc factorial_r.c
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ ./a.out
Ingrese un entero positivo para calcular su factorial: 4
El factorial de 4 es 24
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$

```

3.2 Fibonacci

```

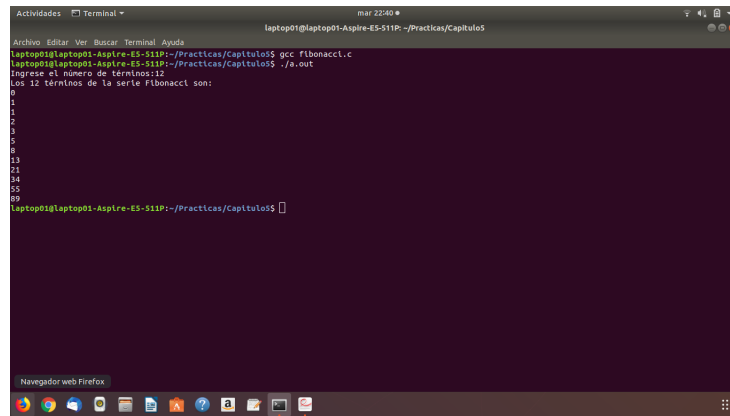
#include<stdio.h>
#include<conio.h>

long fibonacci(int);
int main(){
    int i;

    printf("NUMERO\t FIBONACCI\n\n");
    for(i=0;i<=20;i++)
        printf("%d\t %d\n", i, fibonacci(i));
        getch();
}

long fibonacci(int n)
{
    if(n==0 || n==1)/*Caso base */
        return n;
    else
        return fibonacci(n-2) + fibonacci(n-1);
}

```



A screenshot of a Linux terminal window. The title bar shows 'Actividades' and 'Terminal'. The terminal content shows the user running a C program to calculate the 12th term of the Fibonacci sequence. The output displays the sequence from 0 to 89. The terminal window is titled 'laptop01@laptop01-Aspire-E5-511P: ~/Practicas/Capitulo5'.

```
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ gcc fibonacci.c
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ ./a.out
Ingrese el número de términos:12
Los 12 términos de la serie Fibonacci son:
0
1
1
2
3
5
8
13
21
34
55
89
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$
```

3.3 Máximo Común Divisor

```
#include<stdio.h>

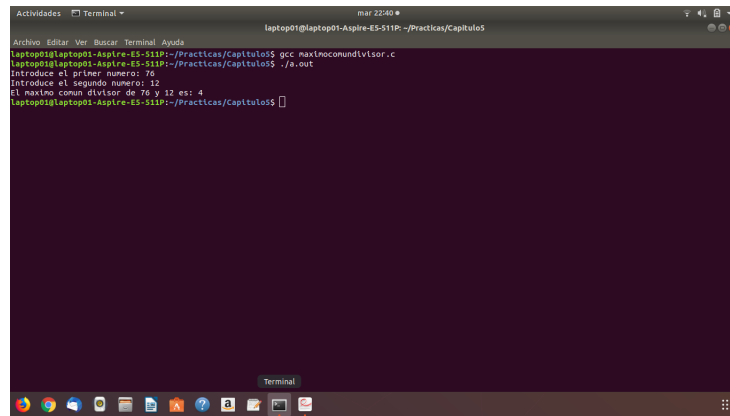
int mcd(int a, int b);

int main(){
    int a,b,resultado;

    printf("Introduce el primer numero: ");
    scanf("%d",&a);
    printf("Introduce el segundo numero: ");
    scanf("%d",&b);
    resultado=mcd(a,b);
    printf("El maximo comun divisor de %d y %d es: %d\n" , a, b, resultado);
    return 0;
}

int mcd(int a, int b){

if (b != 0)
    return mcd(b, a%b);
else
    return a;
}
```



```
Actividades Terminal mar 22:40
laptop01@laptop01-Aspire-E5-511P: ~/Practicas/Capitulo5
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ gcc maxcomundivisor.c
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ ./a.out
Introduce el primer numero: 76
Introduce el segundo numero: 12
El maximo comun divisor de 76 y 12 es: 4
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$
```

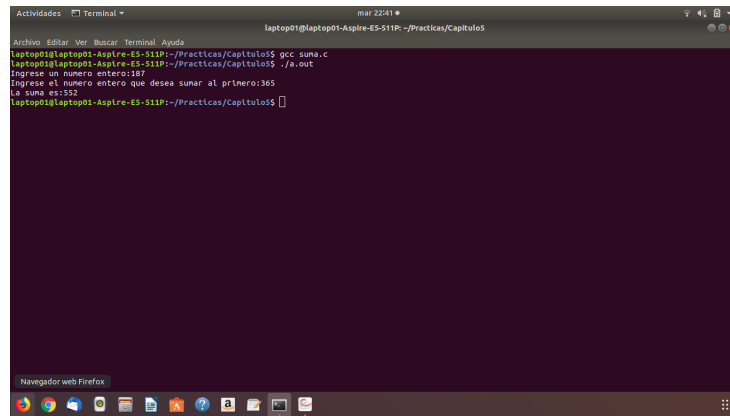
3.4 Suma de dos enteros

```
#include <stdio.h>
int main (){

    int a, b, suma;

    printf ("Ingrese un numero entero:");
    scanf ("%d" , &a) ;
    printf ("Ingrese el numero entero que desea sumar al primero:");
    scanf ("%d", &b);
    suma = a+b;
    printf ("La suma es:%d \n", suma);
    return 0;

}
```



```
Actividades Terminal mar 22:41
laptop01@laptop01-Aspire-E5-511P: ~/Practicas/Capitulo5
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ gcc suma.c
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ ./a.out
Ingrese un numero entero:187
Ingrese el numero entero que desea sumar al primero:365
La suma es:552
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$
```

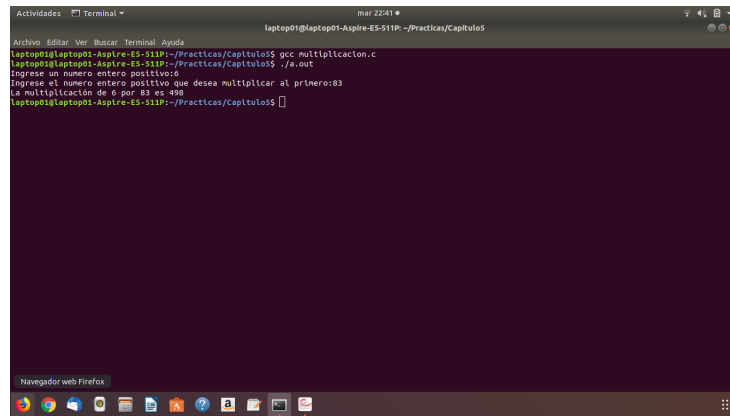
3.5 Multiplicación

```
#include <stdio.h>

int multiplicacion(int a, int b);

int main(){
    int a, b, r;
    printf ("Ingrese un numero entero positivo:");
    scanf ("%d" , &a) ;
    printf ("Ingrese el numero entero positivo que desea multiplicar al primero:");
    scanf ("%d", &b);
    r = multiplicacion(a, b);
    printf("La multiplicación de %d por %d es %d\n", a, b, r);
    return 0;
}

int multiplicacion(int a, int b){
    if (a < b){
        return multiplicacion(b, a);
    }
    else if (b!=0){
        return (a + multiplicacion(a, b - 1));
    }
    else{
        return 0;
    }
}
```



```
Actividades Terminal mar 22:41
laptop01@laptop01-Aspire-E5-511P: ~/Practicas/Capitulo5

laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ gcc multiplicacion.c
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ ./a.out
Ingrese un numero entero positivo:83
Ingrese el numero entero positivo que desea multiplicar al primero:498
La multiplicación de 83 por 498 es 41334
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$
```

3.6 Potencia

```
#include <stdio.h>
```

```
long potencia(long b, int e);
```

```
int main(){
    long b,r;
    int e;

    printf("Ingrese la base:");
    scanf("%ld", &b);
    printf("Ingrese el exponente: ");
    scanf("%d", &e);

    r = potencia(b, e);

    printf("%ld elevado a la %d es: %ld \n", b, e, r);

    return 0;
}
```

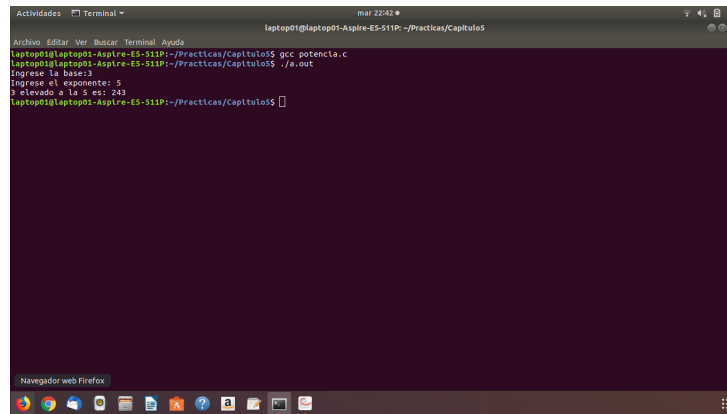
```
long potencia(long b, int e){
    if(e == 0){
        return 1;
    }
    else if(e > 0){
        return b*potencia(b, e-1);
    }
}
```

```

else{
    return 1/potencia(b, - e);
}

}

```



3.7 Torres de Hanoi

```

#include <stdio.h>

void th(int n, char c, char f, char a);

int main(){
    int n;

    printf("Ingrese el numero de discos:");
    scanf("%d", &n);
    printf("La secuencia de movimientos a realizar es :\n");
    th(n, 'A', 'C', 'B');
    return 0;

}

void th(int n, char c, char f, char a){
    if (n == 1){
        printf("\nMueva el disco 1 de %c a %c\n", c, f);
        return;
    }
    th(n - 1, c, a, f);
    printf("\nMueva el disco %d de %c a %c\n", n, c, f);
    th(n-1, a, f, c);

}

```

```
Actividades Terminal mar 22:43
laptop01@laptop01-Aspire-E5-511P: ~/Practicas/Capitulo5

laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ gcc Torres_Hanoi.c
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ ./a.out
Ingrese el número de discos:4
La secuencia de movimientos a realizar es :

Mueva el disco 1 de A a B
Mueva el disco 2 de A a C
Mueva el disco 1 de B a C
Mueva el disco 3 de A a B
Mueva el disco 1 de C a A
Mueva el disco 2 de C a B
Mueva el disco 1 de A a B
Mueva el disco 4 de A a C
Mueva el disco 1 de B a C
Mueva el disco 2 de B a A
Mueva el disco 1 de C a A
Mueva el disco 3 de B a C
Mueva el disco 1 de A a B
Mueva el disco 2 de A a C
Mueva el disco 1 de B a C
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$
```

3.8 *main* recursivo

Cuando ocurre una *main* recursivo, el *main* se llama así mismo hasta cumplir cierta condición. En este caso, se hizo un programa que contara hasta antes de llegar al número 25.

```
#include<stdio.h>
```

```
int main(){
    static int cuenta = 1;
    printf("%d\n", cuenta++);
    if(cuenta == 25)
        return cuenta;
    else
        return main();
}
```

```
Actividades Terminal mar 22:44
laptop01@laptop01-Aspire-E5-511P: ~/Practicas/Capitulo5

laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ gcc mainrecursivo.c
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$ ./a.out
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
laptop01@laptop01-Aspire-E5-511P:~/Practicas/Capitulo5$
```


3.9 Impresión inversa de entradas desde el teclado

```
#include <stdio.h>
#include <string.h>

void invertir(char *x, int i, int f);

int main(){
    char a[100];
    printf("Ingrese una cadena de texto:\n");
    gets(a);
    invertir(a, 0, strlen(a)-1);
    printf("La cadena al revés se lee:\n %s\n", a);
    return 0;
}

void invertir(char *x, int i, int f){
    char c;

    if (i >= f)
        return;
    c = *(x+i);
    *(x+i) = *(x+f);
    *(x+f) = c;
    invertir(x, ++i, --f);
}
```

