

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA



Fundamentos de Programación

Serie II

Nombre del alumno: **EVER IVAN ROSALES GÓMEZ** Grupo: **24**

Número de cuenta: **320216576**

Realizar el análisis (Reconocer datos de entrada, salida, restricciones y procesos), diagrama de flujo, pseudocódigo, programa y pruebas de escritorio para los siguientes puntos:

- 1) Pida dos números al usuario/a, y que los multiplique. Si la multiplicación da un valor menor a 150, se volverán a pedir los números hasta que la multiplicación de ambos tenga una respuesta mayor a 150. Mostrar la respuesta en cada intento.

```
C prog1.c > main()
1  /* Pida dos números al usuario/a, y que los multiplique. Si la multiplicación da un valor menor a 150,
2  se volverán a pedir los números hasta que la multiplicación de ambos tenga una respuesta mayor a 150.
3  Mostrar la respuesta en cada intento
4  ENTRADA: multiplicación de dos valores
5  SALIDA: mencionar si son mayores o menores a 150
6  RESTRICCIONES: solo numeros positivos */
7  #include <stdio.h>
8  #include <locale.h>
9  int num_1, num_2, resultado;
10 int cont = 0;
11 int main () {
12     setlocale(LC_ALL, "ESPAÑOL");
13     do { //USAREMOS UN DO-WHILE PARA PRIMERO HACER LA MULTIPLICACIÓN Y POSTERIORMENTE VER LA CONDICIÓN
14         //ENTONCES LA OPERACIÓN SE LLEVARÁ A CABO HASTA QUE LOS VALORES DEL USUARIO MULTIPLICADOS SEAN MAYORES A 150.
15         printf("\tIngresa el valor 1: ");
16         scanf("%d", &num_1);
17         printf("\n\tIngresa el valor 2: ");
18         scanf("%d", &num_2);
19         resultado = num_1 * num_2;
20         cont++;
21         printf("\nIngresa nuevos valores: ");
22     } while (resultado < 150); //AQUI SE ENCUENTRA LA CONDICIÓN QUE PEDÍA EL PROGRAMA
23     printf("\nLa multiplicación de los valores es %d", resultado); //ESTE MENSAJE APARECE CUANDO LA CONDICIÓN SEA DIFERENTE,
24     //ADEMAS EN ESTA LÍNEA SE USA LA LIBRERÍA LOCALE.H PARA ACENTUAR LA PALABRA MULTIPLICACIÓN.
25     printf("\n\n");
26     return 0;
27 }
```

● Aldair: SERIE DE EJERCICIOS 2 aldair\$ gcc prog1.c -o prog1.out

● Aldair: SERIE DE EJERCICIOS 2 aldair\$./prog1.out

Ingresa el valor 1: 1

Ingresa el valor 2: 1

Ingresa nuevos valores: Ingresa el valor 1: 5

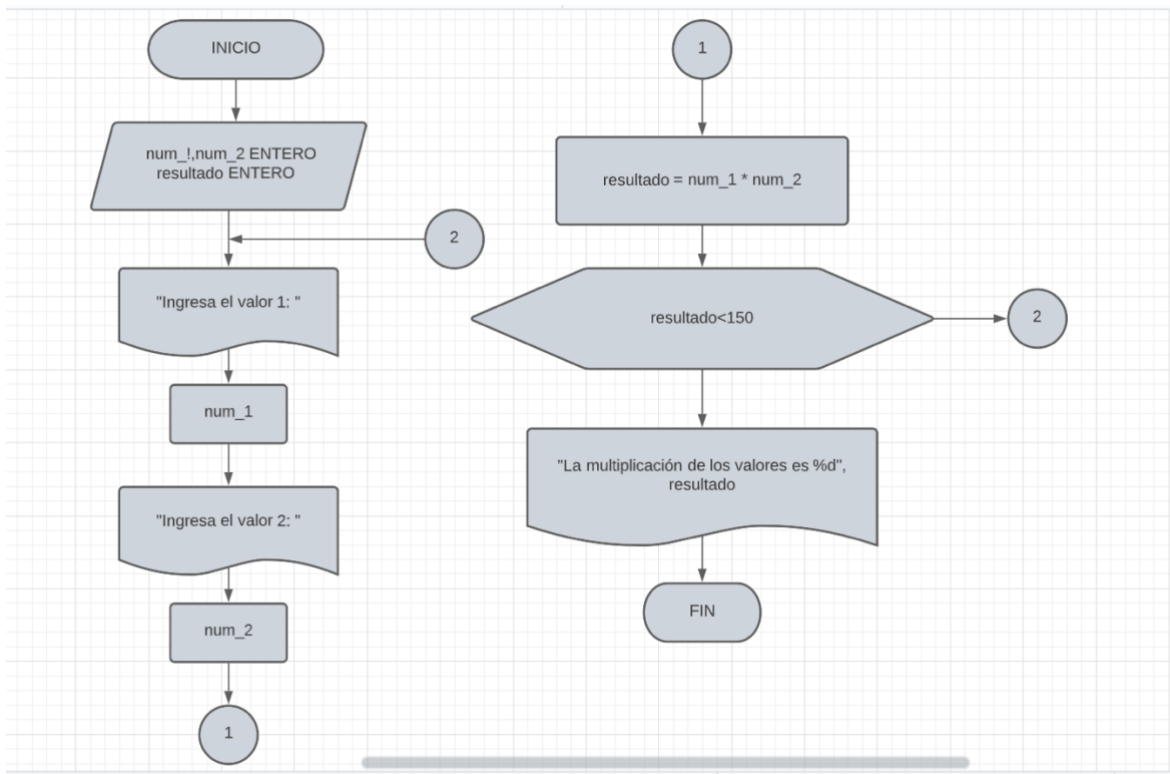
Ingresa el valor 2: 5

Ingresa nuevos valores: Ingresa el valor 1: 10

Ingresa el valor 2: 50

Ingresa nuevos valores:

La multiplicación de los valores es 500



2) Leer un arreglo de tipo carácter con 10 elementos e imprimirlo de forma inversa.

```

C prog2.c > main()
1  /*Leer un arreglo de tipo carácter con 10 elementos e imprimirlo de forma inversa.
2  ENTRADA Un arreglo de 10 elementos
3  SALIDAS Imprimir de forma inversa
4  RESTRICCIONES: Solamente arreglo de tipo caracter
5  */
6  #include <stdio.h>
7  #include <locale.h>
8  char palabra[ ]={'P','r','o','f','a','d','e','E','D','A'}; //DECLARACION DEL ARREGLO
9  int i; //CONTADOR
10
11 int main(){
12     setlocale(LC_ALL,"ESPAÑOL");
13     for(i=0;i<=9;i++)
14         printf("\t%c",palabra[i]);
15     printf("\nLa palabra escrita al revés es "); //AQUÍ SE USA LA LIBRERÍA LOCALE PARA EL ACENTO
16     for(i=9;i>=0;i--)
17         printf("\t%c",palabra[i]);
18     printf("\n");
19     return 0;
20 }
  
```

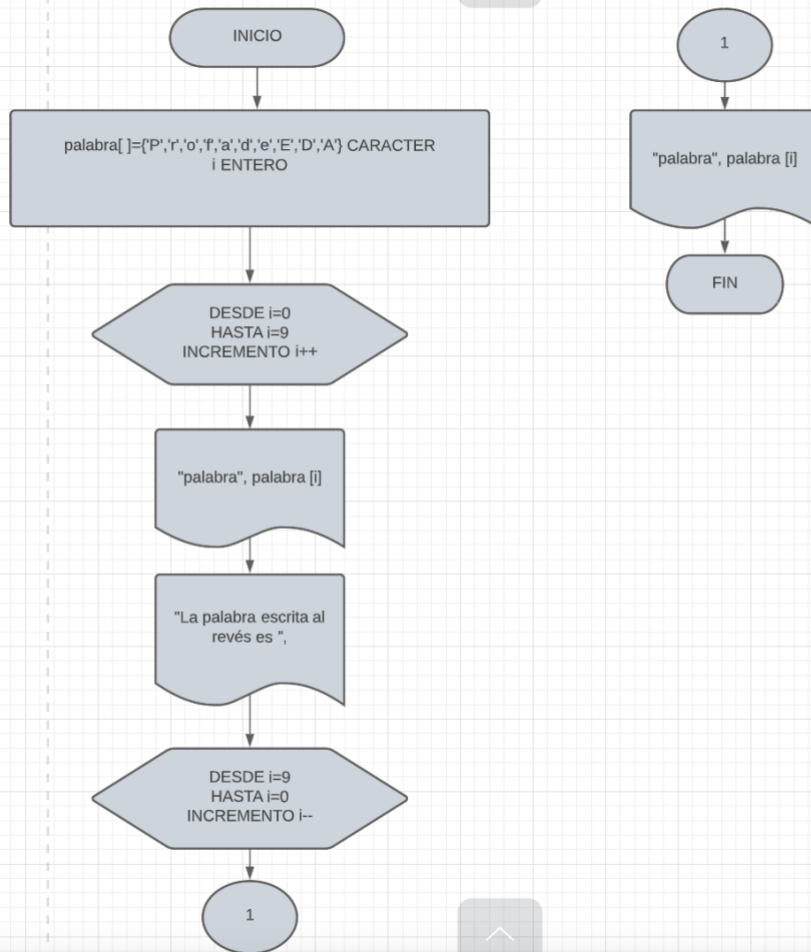
```

Aldair:SERIE DE EJERCICIOS 2 aldair$ gcc prog2.c -o prog2.out
Aldair:SERIE DE EJERCICIOS 2 aldair$ ./prog2.out

```

P
r
o
f
a
d
e
E
D
A

La palabra escrita al revés es A D E e d a f o r P



- 3) Leer dos vectores vectorA y vectorB de 10 elementos de tipo real y obtener el vectorC que almacene la suma del vectorA + vectorB e imprimir cada uno de los 3 vectores al final

```

C prog3.c > main()
1  /*Leer dos vectores vectorA y vectorB de 10 elementos de tipo real y obtener el vectorC
2  que almacene la suma del vectorA + vectorB e imprimir cada uno de los 3 vectores al final
3  ENTRADA dos vectores unidimensionales
4  SALIDAS sumarlos para obtener un tercer vector
5  RESTRICCIONES solamente numeros reales
6  */
7  #include <stdio.h>
8
9  float vectorA[]={1.5,2.5,3.5,4.5,5.5,6.5,7.5,8.5,9.5,10.5,11.5}; //DECLARACION DEL ARREGLO A
10 float vectorB[]={11.5,10.5,9.5,8.5,7.5,6.5,5.5,4.5,3.5,2.5,1.5}; //DECLARACION DEL ARREGLO B
11 float sumaVectorC[10];
12 int i;
13 int main ()
14 {
15     printf("\tVamos a sumar los arreglos establecidos\n");
16     printf("\nLos valores del vector A son: "); //MOSTRANDO LOS VALORES DEL VECTOR A
17     for (i=0;i<10;i++){
18         printf("\n%f",vectorA[i]);
19     }
20     printf("\n");
21     printf("\nLos valores del vector B son: "); //MOSTRANDO LOS VALORES DEL VECTOR B
22     for (i=0;i<10;i++){
23         printf("\n%f",vectorB[i]);
24     }
25     sumaVectorC[i]=vectorA[i]+vectorB[i];
26     printf("\n");
27     printf("\nVector A + Vector B");
28     for (i=0;i<10;i++){
29         printf("\n%f",vectorA[i+1]+vectorB[i+1]); //SE MUESTRA LA SUMA DE LOS VECTORES DE A Y B, EN ESTE CASO TODOS DARÁN 13
30     }
31     printf("\n");
32     return 0;
33 }

```

● Aldair:SERIE DE EJERCICIOS 2 aldair\$ gcc prog3.c -o prog3.out
 ● Aldair:SERIE DE EJERCICIOS 2 aldair\$./prog3.out
 Vamos a sumar los arreglos establecidos

Los valores del vector A son:
 1.500000
 2.500000
 3.500000
 4.500000
 5.500000
 6.500000
 7.500000
 8.500000
 9.500000
 10.500000

Los valores del vector B son:
 11.500000
 10.500000
 9.500000
 8.500000
 7.500000
 6.500000
 5.500000
 4.500000
 3.500000
 2.500000

Vector A + Vector B
 13.000000
 13.000000
 13.000000
 13.000000
 13.000000
 13.000000
 13.000000
 13.000000
 13.000000
 13.000000

○ Aldair:SERIE DE EJERCICIOS 2 aldair\$ █

- 4) Realizar un programa que obtenga la siguiente numeración y la siguiente posición.
 (Utiliza ciclo for).

232

34543

4567654

567898765

```
C prog4.c > main()
1  /*
2  ENTRADA un numero
3  SALIDA numeracion
4  RESTRICCIONES ninguna
5  */
6  #include<stdio.h>
7  int n, c, d, num = 1;
8  int main(){
9
10     printf("\tIngresa un numero\n");
11     scanf("%d",&n);
12     for ( d = 1 ; d <= n ; d++ ){
13         num = d;
14         for ( c = 1 ; c <= n-1 ; c++ )
15             printf(" ");
16         n--;
17         for ( c = 1 ; c <= d ; c++ )
18         {
19             printf("%d", num);
20             num++;
21         }
22         num--;
23         num--;
24     for ( c = 1 ; c < d ; c++){
25         printf("%d", num);
26         num--;
27     }
28     printf("\n");
29
30     return 0;
31 }
```

● Aldair:SERIE DE EJERCICIOS 2 aldair\$ gcc prog4.c -o prog4.out

● Aldair:SERIE DE EJERCICIOS 2 aldair\$./prog4.out

Ingresa un numero

1

1

● Aldair:SERIE DE EJERCICIOS 2 aldair\$ gcc prog4.c -o prog4.out

● Aldair:SERIE DE EJERCICIOS 2 aldair\$./prog4.out

Ingresa un numero

2

1

● Aldair:SERIE DE EJERCICIOS 2 aldair\$./prog4.out

Ingresa un numero

5

1

232

34543

5) Una universidad ofrece un curso que prepara a los estudiantes para el examen de obtención de licencia de corredor de bienes raíces. El año anterior, varios de los estudiantes que completaron el curso presentaron el examen para obtener la licencia. Naturalmente la Universidad desea saber qué resultados obtuvieron sus estudiantes en el examen.

Se nos ha pedido escribir un programa que resuma los resultados recibidos de una lista de 10 estudiantes (Estas calificaciones las deberá ingresar el usuario además del nombre del estudiante). Exhiba un resumen de los resultados de la prueba indicando el número y nombre de los estudiantes que aprobaron y el número de estudiantes que reprobaron. Si más de ocho estudiantes aprueban el examen la Universidad será capaz de aumentar la colegiatura.

```
C prog5.c > ...
1  /*ENTRADA 10 ALUMNOS CON SUS RESPECTIVAS CALIFICACIONES
2  SALIDA IMPRIMIR LA LISTA ADEMÁS SI MAS DE 8 ALUMNOS APROBARON
3  RESTRICCIONES NO HAY CALIFICACIONES NEGATIVAS
4  */
5
6  #include <stdio.h>
7  #include <locale.h>
8  char nombre[10];
9  int calif[1];
10 int i;
11 int main ()
12 {
13     setlocale(LC_ALL,"ESPAÑOL");
14     printf("\t---Bienvenido a la Universidad Corredores---");
15     printf("\n\tREGISTRO DE ALUMNOS");
16     printf("\nIngresa tu nombre separado de tu calificación: ");
17     scanf("%c %d",&nombre[i],&calif[i]);
18
19     printf("\n");
20     for (i=0;i<=10;i++){
21         printf("Nombre: %c con calificación de %d: \n\n",nombre[i],calif[i+1]);
22     }
23
24     return 0;
25 }
26 }

● Aldair:SERIE DE EJERCICIOS 2 aldair$ gcc prog5.c -o prog5.out
● Aldair:SERIE DE EJERCICIOS 2 aldair$ ./prog5.out
---Bienvenido a la Universidad Corredores---
REGISTRO DE ALUMNOS
Ingresa tu nombre separado de tu calificación: ever 9

Nombre: e con calificación de 0:

Nombre:  con calificación de 0:

Nombre:  con calificación de 0:

Nombre:  con calificación de 101:

Nombre:  con calificación de 0:

Nombre:  con calificación de 0:

Nombre:  con calificación de 0:

Nombre:  con calificación de 0:

Nombre:  con calificación de 0:

Nombre:  con calificación de 0:

Nombre:  con calificación de 0:

Nombre:  con calificación de 0:
```

6) Realizar un menú que me muestre los programas realizados previamente (Utiliza funciones).

Notas:

Agregar el desarrollo completo para la solución de los problemas en cada inciso.

Serie contestada correcta y completa equivale a 1 punto sobre segundo parcial, además de la calificación del entregable de esta actividad.

Profesora ya los últimos no supe como resolverlos. Intenté de muchas formas pero no me salía, entendía lo que pedía al igual que la estructura que tuviera el problema, pero no pude