

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

Facultad de Ingeniería, Arquitectura y Diseño



Ingeniería en Computación

Programación Estructurada

ALUMNO: Miguel Angel Portillo Attwell

MATRÍCULA: 370097

GRUPO: 432

PROFESOR: Pedro Nunez Yepiz



Programación Estructurada



INT MAIN:

```
ACTIVIDAD 6 > C: MAPA_PE_ACT6.cpp > ...
1 // MIGUEL ANGEL PORTILLO ATTWELL 370097
2 // FECHA: 12 DE SEPTIEMBRE DEL 2023
3 // DESCRIPCION: PROGRAMA QUE MANEJE UN ESQUELETO DE SELECCION MULTIPLE
4 // QUE FUNCIONE A BASE DE ACTIVIDADES FORMADAS POR FUNCIONES
5 // MAPA_PE_ACT6.cpp
6 #include <stdio.h>
7 #include <stdlib.h>
8 #include <time.h>
9
10 void menu(void);
11 void desendentes(void);
12 void par_impar(void);
13 void min_max(void);
14 void tablas_mult(void);
15
16 int main()
17 {
18     menu();
19     return 0;
20 }
21
22
23 > void menu(void) ...
56
57 > void desendentes(void) ...
73
74 > void par_impar(void) ...
104
105 > void min_max(void) ...
138
139 > void tablas_mult(void) ...
```

MENÚ:

```
ACTIVIDAD 6 > C: MAPA_PE_ACT6.cpp > ...
21 }
22
23 void menu(void)
24 {
25     int operacion;
26     do
27     {
28         printf("M E N U\n");
29         printf("INGRESA CON UN NUMERO LA ACTIVIDAD QUE QUIERES INICIAR\n");
30         printf("[1-NUMEROS DECENDENTES]\n");
31         printf("[2-PAR O IMPAR]\n");
32         printf("[3-NUMERO MIN Y MAX]\n");
33         printf("[4-TABLAS DE MULTIPLICAR]\n");
34         printf("[0-SALIR]\n");
35         printf("ACTIVIDAD: ");
36         scanf("%d", &operacion);
37         switch (operacion)
38         {
39             case 1:
40                 desendentes();
41                 break;
42             case 2:
43                 par_impar();
44                 break;
45             case 3:
46                 min_max();
47                 break;
48             case 4:
49                 tablas_mult();
50                 break;
51             default:
52                 break;
53         }
54     } while (operacion != 0);
55 }
```



1.- Función en C que pida al usuario el valor de n, y desplegar todos los números enteros positivos menores de n en orden descendente.

```
MAPA_PE_ACT5.cpp  MAPA_PE_ACT6.cpp X
ACTIVIDAD 6 > MAPA_PE_ACT6.cpp > ...
47         break;
48     case 4:
49         tablas_mult();
50         break;
51     default:
52         break;
53     }
54 } while (operacion != 0);
55 }
56
57 void desendentes(void)
58 {
59     int n;
60     int i;
61     int j= 0;
62
63     printf("NUMEROS DECENDENTES\n");
64     printf("INGRESE EL VALOR DE N: ");
65     scanf("%d", &n);
66
67     for (i = n - 1; i > 0; i--)
68     {
69         j++;
70         printf("%d.-[%d]\n", j, i);
71     }
72 }
73
74 > void par_impar(void) ...
104
105 > void min_max(void) ...
138
139 > void tablas_mult(void) ...
```



2.- Función en "C" que genere 40 números aleatorios entre el 0 y 200, desplegar los números y la leyenda de cada número si es par o impar , la cantidad de los números pares e impares así como la suma de los números pares o impares.

```
MAPA_PE_ACT5.cpp  MAPA_PE_ACT6.cpp X
ACTIVIDAD 6 > MAPA_PE_ACT6.cpp > ...
71     }
72 }
73
74 void par_impar(void)
75 {
76     int n = 40;
77     int numero;
78     int i;
79     int numpar = 0;
80     int numimpar = 0;
81
82     srand(time(NULL));
83
84     printf("NUMERO PAR E IMPAR\n");
85
86     for ( i = 0; i < n; i++)
87     {
88         numero = rand() % 201;
89
90         if (numero % 2 == 0)
91         {
92             printf("NUMERO %d: [%d]-PAR\n", i + 1, numero);
93             numpar++;
94         }
95         else
96         {
97             printf("NUMERO %d: [%d]-IMPAR\n", i + 1, numero);
98             numimpar++;
99         }
100     }
101     printf("CANTIDAD DE NUMEROS PAR: [%d]\n", numpar);
102     printf("CANTIDAD DE NUMEROS IMPAR: [%d]\n", numimpar);
103 }
104
105 > void min_max(void) ...
138
139 > void tablas_mult(void) ...
```



3.- Función en "C" que genere N (35) cantidad de números (100 -200), desplegar al final el número mayor y el número menor.

```
MAPA_PE_ACT5.cpp  MAPA_PE_ACT6.cpp X
ACTIVIDAD 6 > MAPA_PE_ACT6.cpp > ...
104
105 void min_max(void)
106 {
107     int n = 35;
108     int numero;
109     int min = 200;
110     int max;
111     int i;
112
113     srand(time(NULL));
114
115     printf("NUMERO MINIMO Y MAXIMO\n");
116
117     for ( i = 0; i < n; i++)
118     {
119         numero = rand() % 101 + 100;
120         printf("%d.-%d\n", i+1, numero);
121
122         if (numero < max)
123         {
124             if (numero < min)
125             {
126                 min = numero;
127             }
128         }
129         if (numero > max)
130         {
131             max = numero;
132         }
133     }
134
135     printf("EL NUMERO MENOR ES: [%d]\n", min);
136     printf("EL NUMERO MAYOR ES: [%d]\n", max);
137 }
138
139 > void tablas_mult(void)...
```



4.- Función en "C" que despliegue la tabla de multiplicar de un número dado (número entre el 1 y 20).

```
MAPA_PE_ACT5.cpp  MAPA_PE_ACT6.cpp X
ACTIVIDAD 6 > MAPA_PE_ACT6.cpp > ...
127         }
128     }
129     if (numero > max)
130     {
131         max = numero;
132     }
133 }
134
135 printf("EL NUMERO MENOR ES: [%d]\n", min);
136 printf("EL NUMERO MAYOR ES: [%d]\n", max);
137 }
138
139 void tablas_mult(void)
140 {
141     int i;
142     int n = 10;
143     int tabla;
144     int resultado;
145
146     printf("\n");
147     printf("TABLAS DE MULTIPLICAR\n");
148
149     srand(time(NULL));
150     tabla = rand() % 21;
151
152     for(i = 0; i < n; i++)
153     {
154         int resultado
155         resultado = tabla * (i+1);
156         printf("[%d] x [%d] = [%d]\n", tabla, i+1, resultado);
157     }
158 }
```

GITHUB: [GitHub - MAIKYPORTILLO67/PROGRAMACION_ESTRUCTURADA](https://github.com/MAIKYPORTILLO67/PROGRAMACION_ESTRUCTURADA): Actividades, Reportes y Documentacion de Programacion Estructurada