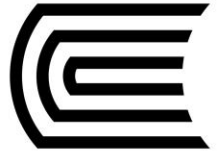




Modularización de programas: tipos de módulos (función y procedimiento) para la creación de menú de opciones



Propósito

Teoría

Reconoce la sintaxis de la creación de menú de opciones en programas con tipos de módulos (función y procedimiento), en el lenguaje de programación



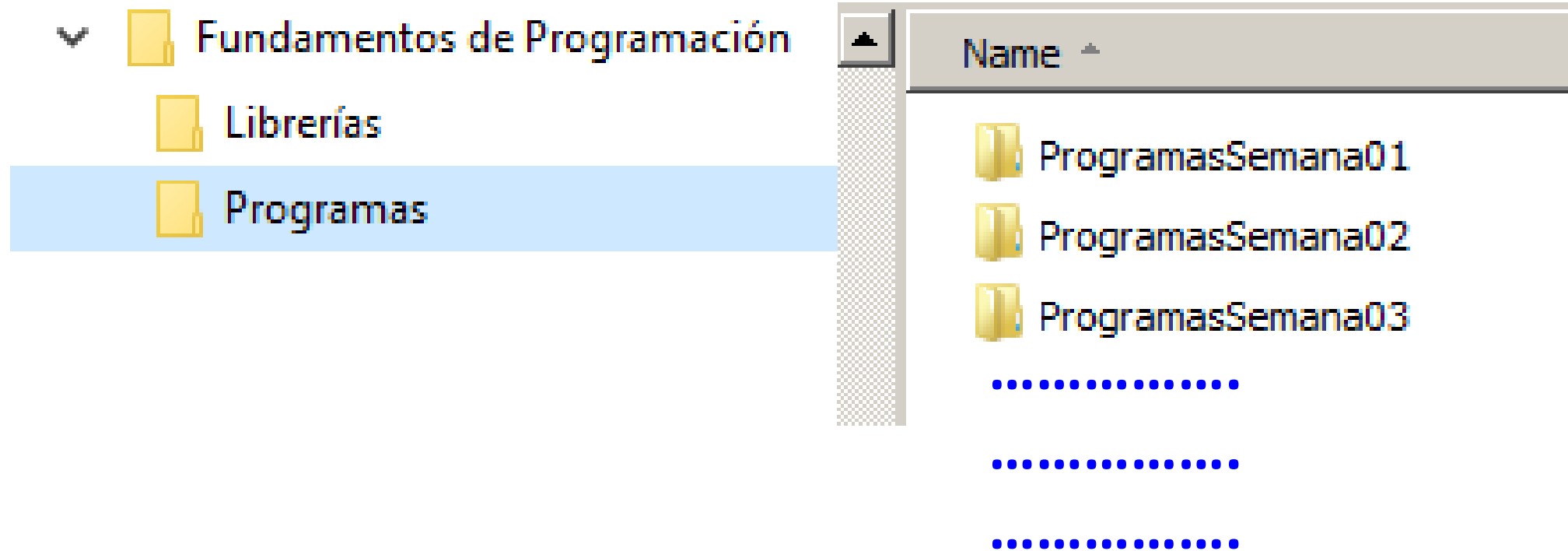
RECUERDA:

Módulo de menú de opciones

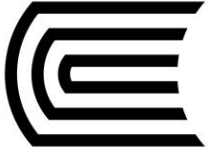
```
79
80 int Menu()
81 {
82     int op;
83
84     do{
85         cout<<"\n\n";
86         cout<<"\t\t ----- MENU -----";
87         cout<<"\n";
88         cout<<"\t\t 1. Procesar."<<endl;
89         cout<<"\t\t 2. Salir."<<endl;
90         cout<<"\n";
91
92         cout<<"\t\t Ingrese opcion (1 o 2): \t";
93         op = leedatoe();
94
95         if(op<1 || op>2)
96             cout<<"\t\t ERROR. Vuelva a Ingresar.";
97
98     }while(op<1 || op>2);
99
100    switch(op)
101    {
102        case 1: {
103            Venta();
104            cout<<"\n";
105
106            }break;
107
108        case 2: {
109            cout<<"\n";
110            cout<<"\t\t GRACIAS..Vuelva propnto..!!";
111            cout<<"\n";
112
113            }break;
114    }
115
116    return op;
117
118 }
119
```

```
120 int main()
121 {
122     int rpt;
123
124     do {
125         rpt = Menu();
126     } while (rpt != 2);
127
128     system("pause");
129
130     return 0;
131 }
132
133
```

Recomendación para crear los programas:



Ahora, elaboremos los programas ejemplos..!



Carpeta: Programas Semana 11

Nombre Proyecto: Proyecto1_VentasMenuModulo

Nombre Programa: Programa1_VentasMenuModulo

```
FruteriasMenuModulos.cpp
1  #include<iostream>
2  using namespace std;
3
4  int leedatoc()
5  {
6      int dato;
7
8      cin>>dato;
9
10     return dato;
11 }
12
13 float leedatof()
14 {
15     float dato;
16
17     cin>>dato;
18
19     return dato;
20 }
21
22 char leedatoc()
23 {
24     char dato;
25
26     cin>>dato;
27
28     return dato;
29 }
30
31
32
```

```
33 void Venta ()
34 {
35     int cant;
36     float precio, dscto, MBruto, MDcto, Mtotal;
37
38     cout<<"\n\n";
39     cout<<"\t\t ---- REALIZAR VENTA ----";
40     cout<<"\n\n";
41     do{
42         cout<<endl<<"\t\t Ingrese cantidad:\t\t";
43         cant = leedatoc();
44         if(cant <=0)
45             cout<<"\t\t ERROR. Debe ser >0.";
46     }while(cant <=0);
47
48     do{
49         cout<<"\t\t Ingrese precio:\t\t";
50         precio = leedatof();
51         if(precio <=0)
52             cout<<"\t\t ERROR. Debe ser >0.";
53     }while(precio <=0);
54
55     if(cant > 100)
56         dscto = 0.4;
57     else
58         if(cant > 25 && cant <= 100 )
59             dscto = 0.2;
60         else
61             if(cant > 10 && cant <= 25 )
62                 dscto = 0.1;
63             else
64                 if(cant <= 10)
65                     dscto = 0;
66
67     MBruto = cant * precio;
68     MDcto = dscto * MBruto;
69     Mtotal = MBruto - MDcto;
70
71     cout<<endl;
72     cout<<"\t\t El monto bruto es:\t\t"<<MBruto<<endl;
73     cout<<"\t\t El monto descuento es:\t\t"<<MDcto<<endl;
74     cout<<"\t\t El monto total es:\t\t"<<Mtotal<<endl;
75
76
77 }
78
```

```
79
80 int Menu()
81 {
82     int op;
83
84     do{
85         cout<<"\n\n";
86         cout<<"\t\t ----- MENU -----";
87         cout<<"\n";
88         cout<<"\t\t 1. Procesar."<<endl;
89         cout<<"\t\t 2. Salir."<<endl;
90         cout<<"\n";
91
92         cout<<"\t\t Ingrese opcion (1 o 2): \t";
93         op = leedatoc();
94
95         if(op<1 || op>2)
96             cout<<"\t\t ERROR. Vuelva a Ingresar.";
97
98     }while(op<1 || op>2);
99
100     switch(op)
101     {
102     case 1: {
103         Venta();
104         cout<<"\n";
105     }break;
106
107     case 2: {
108         cout<<"\n";
109         cout<<"\t\t GRACIAS..Vuelva propnto..!!";
110         cout<<"\n";
111     }break;
112
113     }
114
115     return op;
116
117 }
118
119
```

a. Agregar el módulo
sin argumentos:
Salir()

b. Agregar el módulo
sin argumentos:
Mostrar()
Considerar el uso de
Variables Globales

c. Agregar el módulo
sin argumentos:
MostrarFinal()
Para mostrar contador
y acumulador general.
Considerar el uso de
Variables Globales

```
120 int main()
121 {
122     int rptas;
123
124     do {
125         rptas = Menu();
126     } while (rptas != 2);
127
128     system("pause");
129
130     return 0;
131 }
132
133
```





Carpeta: Programas Semana 11

Nombre Proyecto: Proyecto2_CajeroMenuSubMenuModulo

Nombre Programa: Programa2_CajeroMenuSubMenuModulo

```
1 #include<iostream>
2 using namespace std;
3
4 double saldo = 0, depos, retir;
5 int rpt, op, opc;
6
7 int leedatoe()
8 {
9     int dato;
10    cin >> dato;
11    return dato;
12 }
13 double leedatod()
14 {
15     double dato;
16    cin >> dato;
17    return dato;
18 }
19 void Depositar()
20 {
21    do{
22        cout << "\n\t\tIngrese monto a depositar:\t";
23        depos = leedatod();
24        if (depos <= 0)
25            cout << "\t\t ERROR.Vuelva a ingresar " << endl;
26    }while (depos <= 0);
27
28    saldo = saldo + depos;
29 }
30
31 void Retirar()
32 {
33    do{
34        cout << "\n\t\tIngrese monto a retirar:\t";
35        retir = leedatod();
36        if (retir <= 0)
37            cout << "\t\t ERROR.Vuelva a ingresar " << endl;
38    }while (retir <= 0);
39
40    if (retir <= saldo)
41        saldo = saldo - retir;
42    else
43        cout << "\t\tNo se puede. No tiene saldo." << endl;
44 }
45
46 void VerSaldo()
47 {
48    cout << endl;
49    cout << "\t\tSu saldo es:\t\t" << saldo << endl;
50 }
```

Variables
Globales

```
52 void Salida()
53 {
54     cout << "\n\t\t GRACIAS. Vuelva pronto...!\n";
55 }
56
57 void Procesar() // Sub Menú
58 {
59
60     cout << endl << endl;
61     cout << "\t\t\t-----CAJERO-----\t\t" << endl;
62     cout << endl;
63     cout << "\t\t1.Depositar." << endl;
64     cout << "\t\t2.Retirar." << endl;
65     cout << "\t\t3.Saldo." << endl;
66     cout << "\t\t4.Salir." << endl;
67     cout << endl;
68
69     do {
70         cout << "\t\t Ingrese opcion:\t";
71         opc = leedatoe();
72         if (opc < 1 || opc > 4)
73             cout << "\t\t ERROR.Vuelva a Ingresar";
74     } while (opc < 1 || opc > 4);
75
76     switch (opc)
77     {
78     case 1: {
79         Depositar();
80
81         VerSaldo(); // para que vea la información sin entrar al submenú
82         cout << endl;
83     }break;
84
85     case 2: {
86         Retirar();
87
88         VerSaldo(); // para que vea la información sin entrar al submenú
89         cout << endl;
90     }break;
91
92     case 3: {
93         VerSaldo();
94         cout << endl;
95     }break;
96
97     case 4: {
98         Salida(); // Sale solo del submenú..pero volverá al menú principal
99     }break;
100 }
101 }
```

```
103 int Menu() // Menú principal
104 {
105     do{
106         cout << endl << endl;
107         cout << "\t\t\t-----MENU-----\t\t" << endl;
108         cout << endl;
109         cout << "\t\t1.Procesar." << endl;
110         cout << "\t\t2.Salir." << endl;
111         cout << endl;
112
113         cout << "\t\t Ingrese opcion:\t";
114         op = leedatoe();
115
116         if (op != 1 && op != 2)
117             cout << "\t\t ERROR.Vuelva a Ingresar";
118
119     }while (op != 1 && op != 2);
120
121     switch (op)
122     {
123     case 1: {
124         do{
125             Procesar();
126             cout << endl;
127         } while (opc != 4); // repite el submenú
128     }break;
129
130     case 2: {
131         Salida();
132     }break;
133
134     }
135
136     return op;
137 }
138
139 int main()
140 {
141     do{
142         rpt = Menu();
143     }while (rpt != 2);
144
145     return 0;
146 }
147 }
```



Aprendizaje Basado en Retos: ABR

(Con tu mismo grupo de trabajo)

En el Jamboard Semana 11

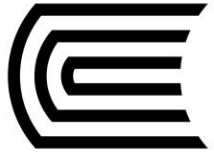
Código: Prototipo-Implementación

1. Iniciar desarrollo del programa usando módulos: menú, lectura de datos, mostrar, solución.

2. El avance (prototipo) enviarlo como imagen al link de jamboard (se dará en clases), una pagina es para cada grupo.

```
1 #include <iostream>
2 using namespace std;
3
4 float leerdatos()
5 {
6     float dato;
7     cin >> dato;
8     return dato;
9 }
10
11 float ecuaciones()
12 {
13     float x,y;
14     cout << "Ingresar valor de x : \n";
15     x = leerdatos(); // usando modulo de lectura float
16
17     if (x==0)
18     {
19         cout << endl << "Ecuacion a usar es y=(x^2)+5\n";
20         y=(x*x)+5;
21     }
22     else
23     {
24         if (x%8==2)
25         {
26             cout << endl << "Ecuacion a usar es y=3x+1\n";
27             y=(3*x)+1;
28         }
29         else
30         {
31             if (x==2)
32             {
33                 cout << endl << "Ecuacion a usar es y=(x^2)+4x+5\n";
34                 y=(x*x)+(4*x)+5;
35             }
36         }
37     }
38     return y;
39 }
40
41 int main()
42 {
43     float res;
44     res=ecuaciones();
45     cout << "Resultado de la funcion es f(y)= " << res << endl;
46     return 0;
47 }
```

3. Revisa la retroalimentación y mejoras que se darán, antes de la presentación final.



Conclusiones

Modularización
de programas:
tipos de
módulos para la
creación de
menú de
opciones

Combina el uso
funciones y
procedimientos.

El módulo
Menú invoca a
otros módulos
para procesar
datos.

El módulo
principal invoca
al módulo
Menú.

Programa

```
57 void Procesar() // Sub Menú
58 {
59
60     cout << endl << endl;
61     cout << "\t\t\t-----CAJERO-----\t\t" << endl;
62     cout << endl;
63     cout << "\t\t1.Depositar." << endl;
64     cout << "\t\t2.Retirar." << endl;
65     cout << "\t\t3.Saldo." << endl;
66     cout << "\t\t4.Salir." << endl;
67     cout << endl;
68
69     do {
70         cout << "\t\t Ingrese opcion:\t";
71         opc = leedatoe();
72         if (opc < 1 || opc > 4)
73             cout << "\t\t ERROR.Vuelva a Ingresar";
74     } while (opc < 1 || opc > 4);
75
76     switch (opc)
77     {
78     case 1: {
79         Depositar();
80         VerSaldo(); // para que vea la información sin entrar al submenú
81         cout << endl;
82     }break;
83
84     case 2: {
85         Retirar();
86         VerSaldo(); // para que vea la información sin entrar al submenú
87         cout << endl;
88     }break;
89
90     case 3: {
91         VerSaldo();
92         cout << endl;
93     }break;
94
95     case 4: {
96         Salida(); // Sale solo del submenú..pero volverá al menú principal
97     }break;
98     }
99
100 }
101
102 int main()
103 {
104     int rpt;
105
106     do {
107         rpt = Menu();
108     } while (rpt != 2);
109     system("pause");
110     return 0;
111 }
```





Propósito

Práctica

Crea programas con menú de opciones y con tipos de módulos (función y procedimiento) en el lenguaje de programación C/ C++.



Desarrollamos la Guía Práctica 11

Guía práctica N°11

Fundamentos de Programación

Tercera Unidad: Módulos para la programación: función y procedimiento

Sección :	Apellidos :
Docente :	Nombres :
	Fecha : / / Duración: 100 min
	Tipo de práctica: Individual (x) Grupal ()

Instrucciones: Lea detenidamente cada enunciado y desarrolle lo solicitado.
Utilizar el Dev C++ para el desarrollo de los siguientes programas

1. Elaborar un programa, para una agencia de viajes turísticos, según la el siguiente menú y tabla. Muestre el pago en soles que realizará según la opción elegida y según el tipo de cambio del día (Ingrese tipo de cambio). Además, crear un módulo, para mostrar la cantidad de Paquetes Cacán que hayan sido elegidos, y el acumulado total de pagos en soles. Puedes usar variables globales, menú principal, sub menú, función o procedimiento, con o sin argumentos, según requiera el caso. Referencia de solución:

Carpeta: Programas Semana 06
Nombre Proyecto: Proyecto5_MenuTurismo
Nombre Programa: Programa5_MenuTurismo

MENÚ DESTINOS TURISTICOS

1. Punta Cana
2. San Andrés
3. Cancún
4. Saltr

Paquete turístico	Precio en US (\$)	Descuento
Punta Cana	700	3.0%
San Andrés	1350	4%
Cancún	2550	4.5%

2. Elaborar un programa, para una empresa que para cada cliente ofrece un descuento, de acuerdo a la tabla siguiente, y de acuerdo al tipo de producto. Ingresar la cantidad y precio del producto seleccionado (Crear menú de opciones de producto) y mostrar el monto bruto, el monto de descuento y el monto total a pagar por cliente. Además, crear u módulo para mostrar la cantidad de veces que se seleccionó cada producto y el acumulado de los montos totales. Puedes usar variables globales, menú principal, sub menú, función o procedimiento, con o sin argumentos, según requiera el caso.

Tipo de Producto	Descuento
[A.] Mouse	8%
[B.] Teclado	10%
[C.] Disco Duro	11%
[D.] Monitor	15%

3. Elaborar un programa, para un restaurante que tiene una lista de bocadillos (menú de opciones). Ingresar el número de unidades consumidas de cada bocadillo seleccionado; y calcular y mostrar el monto total de pago, incluyendo IGV (18%). Además, crear u módulo para mostrar la cantidad de veces que se seleccionó cada bocadillo y el acumulado de los montos totales. Puedes usar variables globales, menú principal, sub menú, función o procedimiento, con o sin argumentos, según requiera el caso.

===== Menú =====
Bocadillo de jamón: 3.50 soles
Patatas fritas: 2.20 soles
Jugo: 3.00 soles
Gaseosa: 3.50 soles

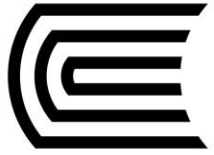
Carpeta: Programas Semana 11
Nombre Proyecto: Proyecto3_AgenciaViajesMenuModulo
Nombre Programa: Programa3_AgenciaViajesMenuModulo

Carpeta: Programas Semana 11
Nombre Proyecto: Proyecto4_ComputadorasMenuModulo
Nombre Programa: Programa4_ComputadorasMenuModulo

Carpeta: Programas Semana 11
Nombre Proyecto: Proyecto5_RestauranteMenuModulo
Nombre Programa: Programa5_RestauranteMenuModulo

Carpeta: Programas Semana 10
Nombre Proyecto: Proyecto8_MasaCorporalMenuModulo
Nombre Programa: Programa8_MasaCorporalMenuModulo

- JOYANES, L. (2008). Fundamentos de Programación. Algoritmos, estructura de datos y objetos, 4ª Edición. Madrid: McGraw-Hill.



Conclusiones

Modularización
de programas:
tipos de
módulos para la
creación de
menú de
opciones

Combina el uso
funciones y
procedimientos.

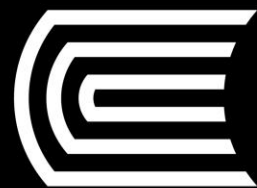
El módulo
Menú invoca a
otros módulos
para procesar
datos.

El módulo
principal invoca
al módulo
Menú.

Programa

```
57 void Procesar() // Sub Menú
58 {
59
60     cout << endl << endl;
61     cout << "\t\t\t-----CAJERO-----\t\t" << endl;
62     cout << endl;
63     cout << "\t\t1.Depositar." << endl;
64     cout << "\t\t2.Retirar." << endl;
65     cout << "\t\t3.Saldo." << endl;
66     cout << "\t\t4.Salir." << endl;
67     cout << endl;
68
69     do {
70         cout << "\t\t Ingrese opcion:\t";
71         opc = leedatoe();
72         if (opc < 1 || opc > 4)
73             cout << "\t\t ERROR.Vuelva a Ingresar";
74     } while (opc < 1 || opc > 4);
75
76     switch (opc)
77     {
78     case 1: {
79         Depositar();
80         VerSaldo(); // para que vea la información sin entrar al submenú
81         cout << endl;
82     }break;
83
84     case 2: {
85         Retirar();
86         VerSaldo(); // para que vea la información sin entrar al submenú
87         cout << endl;
88     }break;
89
90     case 3: {
91         VerSaldo();
92         cout << endl;
93     }break;
94
95     case 4: {
96         Salida(); // Sale solo del submenú..pero volverá al menú principal
97     }break;
98     }
99
100 }
101
102 int main()
103 {
104     int rpt;
105
106     do {
107         rpt = Menu();
108     } while (rpt != 2);
109     system("pause");
110     return 0;
111 }
```





ucontinental.edu.pe