

PROGRAMACIÓN I (PUI)

TRABJO PRÁCTICO 2: MODULARIDAD

Objetivos:

Al finalizar este práctico se espera que los alumnos logren:

- Comprender el concepto de Programación Modular
- Resolver problemas utilizando diseño modular
- Conceptualizar y usar correctamente variables locales y globales y su ámbito o alcance.
- Comprender y utilizar correctamente el concepto de tipos y pasaje de parámetros
- Verificar la solución algorítmica.

Dados los siguientes problemas se pide:

- Diseñar el árbol de jerarquía de módulos aplicando los principios de Programación Modular.
- Expresar en Pseudocódigo las estructuras de datos a utilizar
- Realizar el diagrama de Flujo de las soluciones propuestas, indicando descripción de la tarea, tipo de cada módulo y definición completa de los parámetros necesarios (E-S-E/S).
- Verificar el correcto funcionamiento de cada uno de los módulos.

1) El supermercado “**Día**” desea que se le confeccione un programa que le brinde las siguientes opciones:

- 1.- Actualización de precios
- 2.- Informe de Productos
- 3.- Finalizar el programa

Para ello dispone de la siguiente información:

- ✓ El arreglo **P[DIM]**, de tipo registro contiene los siguientes datos: Código Producto, Descripción, Cantidad existente y Precio.
- ✓ En el módulo *Actualización de precios*: se lee un código de producto y el porcentaje a incrementar, si el producto existe deberá actualizar su precio, en caso contrario mostrará el mensaje “*Producto Inexistente*”
- ✓ En el módulo *Informe de productos*: realizar un listado con el siguiente formato de aquellos productos cuyo monto en stock sea mayor que \$100.000

Informe de productos		
Código	Descripción	Monto en stock
101	HARINA	550.000
123	FIDEOS	150.000
154	ACEITE	320.000

- ✓ A los efectos de la elaboración de la solución complete el programa reducido mostrado a continuación:

PROGRAMA Supermercado DIA

VARIABLES: Opción, P, DIM

REPEAT

MOSTRAR “1.- Actualización de Precios”

MOSTRAR “2.- Informe de Productos”

MOSTRAR “3.- Finalizar el Programa”

LEER Opcion

CASE (Opcion)

Opcion = 1: ActualizaPrecio(P, DIM)

Opcion = 2: InformeProductos (P, DIM)

ENDCASE

UNTIL Opcion = 3

FIN

PROCESO ActualizaPrecio (P: , DIMP:)

// Actualiza el precio de un producto ingresado, si se encuentra

FIN

PROCESO InformeProductos (P: , DIMP:)

// Muestra un listado de los productos que cumplen con la condición especificada

FIN

PROCESO Ordenar (P: , DIMP)

// ordena el arreglo P en forma ascendente por código de producto)

FIN

Función Busco (CodigoProd: , P: , DIMP:)

// Función que busca el código de producto en el arreglo P y retorna la posición encontrada ó -1 si no existe)

FIN

Resolución en Pseudocodigo:

```
Producto:Registro
    codP: entero
    desc: cadena
    cantE: entero
    precio: real
FinRegistro
```

```
Proceso ActualizaPrecio(P,DIMP)
    Definir cod_prod, i, b como Entero
    Definir porI como Real
    Escribir "Ingresa el código de un producto"
    Leer cod_prod
    Escribir "Porcentaje a incrementar"
    Leer porI
    i = 0
    b = 0
    Mientras (i < DIMP y b = 0) Hacer
        Si (P[i].codP = cod_prod) Entonces
            porI = (porI / 100) + 1
            P[i].codP = P[i].codP * porI
            b = 1
        FinSi
        i = i + 1
    FinMientras
    Si (b = 0) Entonces
        Escribir "Producto Inexistente"
    FinSi
FinProceso
```

```
Proceso InformeProducto(P, DIMP)
    Definir i como Entero
    Definir montos como Real
    i = 0
    Escribir "Informe de Productos"
    Escribir "Codigo      Descripcion      Monto en Stock"
    Mientras (i < DIMP) Hacer
        montos = P[i].precio * P[i].cantE
        Si (montos > 100000) Entonces
            Escribir P[i].codP, P[i].desc, montos
        FinSi
        i = i + 1
    FinMientras
FinProceso
```

```
Proceso Ordenar(P, DIMP)
```

```

Definir i, b, cod, can como Entero
Definir cade como Cadena
Definir pre como Real
i = 0
Repetir
    b = 0
    i = 0
    Mientras (i < DIMP - 1) Hacer
        Si (P[i].codP > P[i + 1].codP) Entonces
            cod = P[i].codP
            P[i].codP = P[i + 1].codP
            P[i + 1].codP = cod
            cade = P[i].desc
            P[i].desc = P[i + 1].desc
            P[i + 1].desc = cade
            can = P[i].cantE
            P[i].cantE = P[i + 1].cantE
            P[i + 1].cantE = can
            pre = P[i].precio
            P[i].precio = P[i + 1].precio
            P[i + 1].precio = pre
            b = 1
        FinSi
        i = i + 1
    FinMientras
Hasta (b = 0)
FinProceso

```

```

Funcion resultado = Busco(CodigoProd, P, DIMP)
    Definir i, b como entero
    i = 0
    b = 0
    Mientras (i < DIMP y b = 0) Hacer
        Si (CodigoProd = P[i].codP) Entonces
            resultado = i
            b = 1
        SiNo
            i = i + 1
        FinSi
    FinMientras
    Si (b = 0) Entonces
        resultado = -1
    FinSi
FinFuncion

```

```

Programa SuperMercado_Dia
    Definir Opcion, Dim, cp como entero
    Escribir "Ingrese la Dimension"
    Leer Dim
    P[Dim]= Producto
    Repetir
        Escribir "1.- Actualizacion de Producto"
        Escribir "2.- Informe de Productos"
        Escribir "3.- Ordenar Productos"
        Escribir "4.- Buscar Producto"
        Leer Opcion
        Según (Opcion) Hacer
            caso 1:
                ActualizaPrecio(P, Dim)
            caso 2:
                InformeProductos(P, Dim)
            Caso 3:

```

```

        Ordenar(P, Dim)
    caso 4:
        Escribir "Ingresa el código de producto a Buscar"
        Leer cp
        Escribir "Se encuentra en la posición: ", Busco(cp, P, Dim)
    FinSegun
Hasta (Opcion = 5)
FinPrograma

```

2) Una empresa dedicada al alquiler de automóviles necesita determinar el monto que deben pagar sus clientes por el alquiler de los mismos. Para ello, en primer término, deberá leer el ingreso de la fecha actual (representada por el día, el mes y el año) y controlar si es correcta mediante el uso de una función lógica, en caso de ser incorrecta deberá ser reingresada hasta que la misma sea una fecha valida. Posteriormente se deberá leer el Código del Auto que se alquiló y la distancia que recorrió el cliente con el mismo. El monto por alquiler será establecido en función de un Monto Base, de la cantidad de días de alquiler y de un descuento. Para ello, dispone de un arreglo **Autos(N)** de tipo registro, con la siguiente información de todos los automóviles que dispone: Código del Auto, Marca y Categoría del Auto { Chico, Mediano, Grande}.

A su vez dispone de un arreglo **Alquiler(N)**, de tipo registro, en el cual se almacena la información de los automóviles que se encuentran alquilados: Código del Auto y Fecha Alquiler (Día, Mes y Año) en que fue alquilado el automóvil, se considera que el año de alquiler es el mismo que el actual.

El monto que se debe pagar se lo determina de la siguiente manera:

Monto Base = TarifaBase + DistanciaRecorrida x Precio por Km Recorrido

Monto Alquiler = Monto Base * Cant. de días – Descuento

En donde las tarifas vigentes se encuentran establecidas en la siguiente tabla:

Categoría Auto	Tarifa Base	Precio por Km Recorrido
Chico	300	100
Mediano	600	200
Grande	900	300

Y los descuentos se los aplican de acuerdo a las siguientes reglas:

- Si la distancia recorrida es > 300 Km y <= 1000Km se descontará un 10%
- Si la distancia recorrida es > 1000Km, se le descontará un 15%
- Mientras que si no sobrepasa los 300 Km, no se le aplicará ningún descuento

Resolucion en Pseudocodigo:

Fecha:Registro

 dia: numerico

 mes: numerico

 año: numerico

FinRegistro

Auto:Registro

 codA: entero

 marca: cadena

 catA: cadena

FinRegistro

Alquiler:Registro

 codgA: entero

 fecha: Fecha

FinRegistro

Funcion resultado = CompFecha(dia,mes,año)

 Si((dia> 0 y dia < 32) y (mes> 0 y mes < 13) y (año > 1978 y año < 2023)) Entonces

 resultado = 1

 SiNo

```

        resultado = 0
    FinSi
FinFuncion

Funcion resultado = CantDias(Al, Dim, CodA, dia,mes,año)
    //Se considera que el año de alquiler es el mismo que el actual
    Definir i, b, cal_m como Entero
    i = 0
    b = 0
    resultado = 0
    Mientras(i < Dim y b = 0) Hacer
        Si (CodA = Al[i].codgA) Entonces
            cal_m = (mes - Al[i].fecha.mes) * 31
            resultado = 31 - Al[i].fecha.dia + dia + cal_m
            b = 1
        SiNo
            i = i + 1
        FinSi
    FinMientras
FinFuncion

Programa Alquiler_Auto
    Definir N,día,mes,año, resultado, código_auto, opc_cat, cantD como
Entero
    Definir dist, monto_base, montoAl, descuento como Real
    Leer N
    Auto[N] = Auto
    Al[N] = Alquiler
    Leer día
    Leer año
    Leer año
    resultado = -1
    Mientras(resultado <> 1) Hacer
        Escribir "Ingrese una fecha valida"
        Leer día
        Leer año
        Leer año
        resultado = CompFecha(día, mes, año)
    FinMientras
    Escribir "Ingrese el código de Auto"
    Leer código_auto
    Escribir "Ingrese la distancia recorrida en KM"
    Leer dist
    Escribir "Seleccione la Categoria"
    Escribir "1.- Chico"
    Escribir "2.- Mediano"
    Escribir "3.- Grande"
    Leer opc_cat
    descuento = 0
    Si(dist > 300 y dist <= 1000) Entonces
        descuento = 10
    SiNo
        Si(dist > 1000) Entonces
            descuento = 15
        FinSi
    FinSi

```

```

cantD = CantDias(Al, N, código_auto, dia, mes, año)
Según (opc_cat) Hacer
  Case 1:
    monto_base = 300 + 100 * dist
    montoAl = monto_base * cantD
    montoAl = montoAl - montoAl * descuento /100

  case 2:
    monto_base = 600 + 200 * dist
    montoAl = monto_base * cantD
    montoAl = montoAl - montoAl * descuento /100

  case 3:
    monto_base = 900 + 300 * dist
    montoAl = monto_base * cantD
    montoAl = montoAl - montoAl * descuento /100

FinSegun
FinPrograma

```

3) Una empresa que se dedica a comercializar sábanas y toallas desea organizar las entregas a sus clientes. Cada cliente puede realizar varias compras en distintas fechas
 Nota: en color **rojo** se indica el nombre que se le dará a la variable.

CLIENTES

DNI	Número de Factura	Monto Facturado	Fecha de compra
DNI	NF	MF	FC
32.823.387	178	10000	26/08/2022
14.823.945	234	46000	26/08/2022
32.823.387	145	48000	27/08/2022

FACTURAS

Número Factura	Lugar de entrega	Fecha de entrega
NF	LE	FE
178	SDE	02/09/22
234	LB	05/09/22
145	SDE	01/09/22

Las tareas que debe realizar el programa, mediante el menú correspondiente son:

- Buscar por Número de factura e informar lugar y fecha de entrega. Si no existe mostrar el mensaje "No se encontró".
- Dado un cliente determinado (DNI) informar el total facturado.
- Listar los DNI con la totalidad de compras realizadas según el siguiente formato:

DNI: 32.823. 387

N° Factura	Monto a pagar	Fecha entrega
178	10000	02/09/22
145	46000	01/09/22
<i>Total por cliente</i>		56000

DNI: 14.823.945

N° Factura	Monto a pagar	Fecha entrega
234	40000	05/09/22
<i>Total por cliente</i>		40000

<i>Total General</i>	96000
----------------------	-------

Resolución en Pseudocódigo:

```
Fecha:Registro
    dia: entero
    mes: entero
    año: entero
FinRegistro
```

```
Cliente:Registro
    DNI: entero
    NF: entero
    MF: entero
    FC: Fecha
FinRegistro
```

```
Facturas:Registro
    NF: entero
    LE: cadena
    FE: Fecha
FinRegistro
```

```
Proceso BuscarFactura(Facturas, DF,NF)
    Definir i,b como Entero
    i = 0
    b = 0
    Mientras(i < DF y b = 0) Hacer
        Si(NF = Facturas[i].NF) Entonces
            Escribir "Lugar de Entrega: ", Facturas[i].LE
            Escribir "Fecha de Entrega: ", Facturas[i].FE
            b = 1
        FinSi
        i = i + 1
    FinMientras
    Si (b = 0) Entonces
        Escribir "No se encontro"
    FinSi
FinProceso
```

```
Proceso DetalleCliente(Clientes, DC, DNI)
    Definir i, suma como Entero
    suma = 0
    i = 0
    Mientras(i < DC) Hacer
        Si(DNI = Clientes[i].DNI) Entonces
            suma = suma + Clientes[i].MF
        FinSi
        i = i + 1
    FinMientras
    Escribir "Total Facturado: ", suma
FinProceso
```

```
Proceso DetalleCompra(Clientes,DC,Facturas,DF)
    Definir i, j, suma como Entero
    suma = 0
    i = 0
    Escribir "DNI: ", DNI
```

```

    Escribir "N° Factura"           Monto a pagar           Fecha entrega"

Mientras(i < DF) Hacer
    j = 0
    Mientras(j < DC) Hacer
        Si(DNI = Clientes[j].DNI y Clientes[j].NF = Facturas[i].NF)
Entonces
        suma = suma + Clientes[j].MF
        Escribir Clientes[j].NF, Clientes[j].MF, Clientes[j].FE
    FinSi
    j = j + 1
FinMientras
i = i + 1
FinMientras
Escribir "Total por Cliente: ", suma
FinProceso

Programa Ej_3
    Definir DC,DF, num_f, d_dni, opcion como Entero
    Leer DC
    Leer DF
    Clientes[DC] = Clientes
    Facutas[DF] = Facturas
    Repetir
        Escribir "1.- Buscar Numero de Factura"
        Escribir "2.- Informe del Total Factura de un Cliente"
        Escribir "3.- Detalle de los Clientes"
        Escribir "4.- Salir"
        Leer opción
        Según(opción) Hacer
            Caso 1:
                Escribir "Ingrese el Numero de Factura a buscar"
                Leer num_f
                BuscarFactura(Facturas, DF, num_f)
            Caso 2:
                Escribir "Ingrese un DNI"
                Leer d_dni
                DetalleCliente(Clientes, DC, d_dni)
            Caso 3:
                DetalleCompra(Clientes,DC,Facturas,DF)
        FinSegun
    Hasta (opcion = 4)
FinPrograma

```

4) Una empresa que vende materiales de construcción desea organizar las entregas a sus clientes por compras en efectivo. Cada cliente puede tener varias notas de pedido

CLIENTES

DNI	Celular	Teléfono Fijo	Número de nota de pedido	Monto a pagar	Fecha pedido
DNI	TC	TF	NNP	MP	FP
12.387.823	44	4223637	178	20000	26/08/22
17.823.945	70	4275860	234	50000	26/08/22

12.387.823	51	4223637	145	38000	27/08/22
------------	----	---------	-----	-------	----------

PEDIDOS

Número de nota de pedido	Domicilio de entrega	Fecha de entrega
NNP	DE	FE
178	Belgrano Sud 1912	01/09/22
145	Belgrano Sud 1912	05/09/22
234	Belgrano Sud 1912	02/09/22

Las tareas que debe realizar el programa, mediante el menú correspondiente son:

- Buscar por DNI, si existe el CLIENTE informar ambos números de teléfono, caso contrario mostrar el mensaje "No se encontró".
- Dado un cliente determinado informar su monto total de compras.
- Listar los DNI con la totalidad de pedidos realizados según el siguiente Formato:

DNI	12.387.823	Monto a pagar	Fecha de entrega
Nota pedido Nro		20000	01/09/22
178			
145		38000	05/09/22
<i>Total por cliente</i>		58000	

DNI	17.823.945	Monto a pagar	Fecha de entrega
Nota de pedido Nro			
234		50000	02/09/22
<i>Total por cliente</i>		50000	
<i>Total General</i>		108000	

Resolución Pseudocodigo:

```

Fecha:Registro
    dia: entero
    mes: entero
    año: entero
FinRegistro

```

```

Cliente:Registro
    DNI: entero
    TC: entero
    TF: entero
    NNP: entero
    MP: entero
    FP: Fecha
FinRegistro

```

```

Pedidos:Registro
    NNP: entero
    DE: cadena
    FE: Fecha
FinRegistro

```

```

Proceso BuscarFactura(Cliente, DC, DNI)
    Definir i,b como Entero
    i = 0
    b = 0

```

```

Mientras(i < DC y b = 0) Hacer
    Si(NF = Facturas[i].NF) Entonces
        Escribir "Celular: ", Cliente[i].TC
        Escribir "Telefono Fijo: ", Cliente[i].TF
        b = 1
    FinSi
    i = i + 1
FinMientras
Si (b = 0) Entonces
    Escribir "No se encontro"
FinSi
FinProceso

```

```

Proceso DetalleCliente(Clientes, DC, DNI)
    Definir i, suma como Entero
    suma = 0
    i = 0
    Mientras(i < DC) Hacer
        Si(DNI = Clientes[i].DNI) Entonces
            suma = suma + Clientes[i].MP
        FinSi
        i = i + 1
    FinMientras
    Escribir "Monto Total de Compras: ", suma
FinProceso

```

```

Programa Ej_4
    Definir DC, DP, opción, dni como Entero
    Leer DC
    Leer DP
    C[DC] = Clientes
    P[DP] = Pedidos
    Repetir
        Escribir "1.- Buscar Cliente"
        Escribir "2.- Informar monto total de cliente"
        Escribir "3.- Lista de Clientes"
        Escribir "4.- Salir"
        Leer opción
        Según(opción)Hacer
            Caso 1:
                Leer dni
                BuscarFactura(C, DC,dni)
            Caso 2:
                Leer dni
                DetalleCliente(C, DC, DNI)
            Caso 3:
                FinSegun
        Hasta (opcion = 4)
    FinPrograma

```

5) El consorcio de departamentos del edificio **"Independencia"** desea administrar en forma computarizada el cobro de las expensas, para ello necesita que se confeccione un programa que le brinde las siguientes opciones:

- 1.-Consulta Deuda Expensas
- 2.- Informe Anual de Expensas Adeudadas.
- 3.- Finalizar el programa

Para ello dispone de la información necesaria en las siguientes estructuras de datos:

La matriz **EXPENSAS(50x13)**, la cual contiene el importe mensual adeudado durante el año, por el pago de las expensas de los propietarios de los departamentos, y el arreglo de tipo registro **DPTOS(50)**, en el cual se almacena la siguiente información de cada uno de los departamentos del edificio: *Nro de Unidad, Piso, Dpto, Propietario, Mts2* .

EXPENSAS(50 x 13)

Nro Unidad	Enero	Febrero	Marzo		Octubre	Noviembre	Diciembre
101	0	0	0	0	0	0
102	2300	2450	2600	0	0	0
302	0	0	2650	0	0	0
204	0	2450	2600	0	0	0
201	4200	4300	4500	0	0	0
.....

Nota: Si el importe de las expensas contiene un cero (0), significa que ese mes no tiene deuda.

DPTOS [50]

Nro. Unid	Piso	Dpto	Propietario	Mts.	Nro. Unid	Piso	Dpto	Propietario	Mts.	Nro. Unid	Piso	Dpto	Propietario	Mts.
101	1	C	Luna, María	75	302	3	B	López, Manuel	95	204	2	D	Pérez, Carlos	95

- ✓ **Consulta Deuda:** Ingresar Mes actual y Nro. Unidad y mostrar en el caso de que el Nro de Unidad exista, el ejmplo mostrado a continuación. En caso de que el Número de Unidad no exista, imprimir el mensaje de **"Nro. de Unidad Inexistente"**

Ejemplo:

*Si el **Nro de Unidad** ingresado fuese el **204**, y el mes actual: **09**, la salida resultante sería:*

Nro de Unidad: **204** Piso: **2** Dpto: **D**

Propietario: **Pérez, Carlos**

Total de Meses Adeudados: **2**

Total Importe Adeudado: **5050**

- ✓ **Informe Expensas:** Imprimir un informe discriminado por piso de las expensas adeudadas, debiendo el mismo cumplir con el siguiente formato:

Ejemplo:

INFORME ANUAL DE EXPENSAS ADEUDADAS

Piso. 1

Unidad	Propietario	Cant Mes Adeudados	Importe
101	Luna, María	0	0
102	Cianferoni, Sergio	3	7350
			Total: 7350

Piso. 2

Unidad	Propietario	Cant Mes Adeudados	Importe
202	Manfredi, Sonia	2	2450
204	Pérez, Carlos	2	5050
201	Velez, Juan	3	13000
			Total: 20500

Piso. 3

Unidad	Propietario	Cant Mes Adeudados	Importe
302	López, Manuel	2	2650
306	Rojas, Carolina	3	2450
303	Méndez, Dana	1	2300
			Total: 7400

Total General de la deuda

35250

Departamentos: Registro

NroUnidad = Entero

Piso = Entero

Dpto = Char

Propietario = Alfanumérico

Mts2 = Entero

FinRegistro

DPTOS[50] = Departamentos

Proceso resultado = ConsultaDeuda(DPTOS,EXPENSAS,Dim,A,B)

Definir MesAct, NumU como Entero

Dimension resultado[2]

b = 0

i = 0

j = 0

k = 1

MesAd = 0

TImporte = 0

Escribir "Ingrese el mes actual: "

Leer MesAct

Escribir "Ingrese el numero de unidad: "

Leer NumU

Mientras (i < A) && (b==0) Hacer

Si (NumU == DPTOS[i].NroUnidad) Entonces

b = 1

Escribir "Nro de Unidad: ", DEPTOS[i].NroUnidad

Escribir "Piso: ", DEPTOS[i].Piso

Escribir "Dpto: ", DEPTOS[i].Dpto

Escribir "Propietario: ", DEPTOS[i].Propietario

Mientras (j < A) Hacer

Si (NumU == EXPENSAS[j][0]) Entonces

Mientras (k < 13) Entonces

Si (EXPENSAS[j][k] > 0) Entonces

MesAd = MesAd + 1

TImporte = TImporte + EXPENSAS[j][k]

```

                                FinSi
                                k = k + 1
                                FinMientras
                                FinSi
                                j = j + 1
                                FinMientras
                                Escribir "Total de Meses Adeudados: ", MesAd
                                Escribir "Total de Importe Adeudado: ", TImporte
                                FinSi
                                i = i + 1
                                FinMientras

                                Si (b==0) Entonces
                                    Escribir "Nro. de Unidad Inexistente"
                                Sino
                                    resultado[0] = MesAd
                                    resultado[1] = TImporte

FinProceso

Proceso InformeExpensas (DPTOS, EXPENSAS, Dim, A, B)
    i = 1
    Escribir "INFORME ANUAL DE EXPENSAS ADEUDADAS"
    Mientras (i < 10) Hacer
        Escribir "Piso ", i
        Mientras (j < A) Hacer
            Si (DEPTOS[j].Piso == i)
                Escribir "Unidad" "Propietario" "Cantidad Mes Adeudados"
                "Importe"
                Escribir DEPTOS[j].NroUnidad, DEPTOS[j].Propietario,

Programa Expensas_Independencia
    Definir NroUnidad, Piso, Mts2, Opc, Dim, EXPENSAS como Entero
    Definir Dpto como Caracter
    Definir Propietario como Cadena
    Dim = 50, A = 50, B = 13
    Dimension EXPENSAS[A][B]
    Registro DPTOS[Dim]: Departamentos

    Repetir
        Escribir "----- MENU DE OPCIONES -----"
        Escribir " 1) Consulta Deuda Expensas "
        Escribir " 2) Informe Anual de Expensas Adeudadas "
        Escribir " 3) Finalizar el programa "
        Escribir " Introduzca la opcion (1-3): "
        Leer Opcion

        Segun (Opcion) Hacer
            Caso 1:
                ConsultaDeuda (DPTOS, EXPENSAS, Dim, A, B)
            Caso 2:
                InformeExpensas (DPTOS, EXPENSAS, Dim, A, B)
        FinSegun
        Hasta Que (Opcion = 3)
    FinPrograma

```