

**“Año de la unidad, la paz y el desarrollo”**

**GUIA DE LABORATORIO III**

**TEMA: ESTRUCTURA DE CONTROL SELECTIVA ANIDADA**

**DATOS INFORMATIVOS**

<b>1.1. Nombre de la asignatura</b>	<b>: Algoritmos</b>
<b>1.2. Semestre académico</b>	<b>: 2023-0</b>
<b>1.3. Ciclo de estudios</b>	<b>: I</b>
<b>1.4. Área de estudios</b>	<b>: Específicos</b>
<b>1.5. Nombre del docente</b>	<b>: M.Sc. Luis A. Holgado Apaza</b>
<b>1.6. Nombre del Alumno</b>	<b>: Angel Patrick Areque Aedo</b>

**INTRODUCCION.**

En esta oportunidad vengo a presentar los 5 ejercicios de la guía de laboratorio número 3 del curso de algoritmo de la Carrera profesional de ingeniería de sistemas e informática, en las siguientes paginas daré a conocer el análisis, diagrama de flujo, pseudocódigo y codificación de cada uno de los problemas propuestos por el docente.

# “Año de la unidad, la paz y el desarrollo”

- 1) En una empresa, los salarios de los empleados se van aumentar según el sueldo actual:

Sueldo actual	Aumento%
0 a 800	20
801 a 1000	10
1001 a 1500	5
Más de 1500	0

Desarrolle un programa que solicite el sueldo actual de un empleado, calcule y visualice el nuevo sueldo.

Entrada	Proceso	Salida	
Sueldo_A		Aumento	Sueldo_fin
750		150	900
930		93	1023
1250		62.5	1312.5
1800		0	1800

## 1.2. Modelo

$$\text{Sueldo\_fin} = \begin{cases} \text{Aumento} = 0.2 & , ( \text{sueldo\_A} > 0 ) \text{ y } ( \text{sueldo\_A} < 800 ) \\ \text{Aumento} = 0.1 & , ( \text{sueldo\_A} > 801 ) \text{ y } ( \text{sueldo\_A} < 1000 ) \\ \text{Aumento} = 0.05 & , ( \text{sueldo\_A} > 1001 ) \text{ y } ( \text{sueldo\_A} < 1500 ) \\ \text{Aumento} = 0 & , \text{sueldo\_A} > 1500 \end{cases}$$

## 1.3. Especificación del algoritmo

### 1.3.1. Diccionario de variables

#### Variables de entrada

Sueldo\_A {representa el sueldo del mes del empleado }: tipo Entero

#### Variables de proceso

Aumento {representa el aumento segun el sueldo del empleado }tipo real.

#### Variables de salida

Sueldo\_fin {representa el sueldo del empleado mas el aumento }: tipo real.

### 1.3.2. Pre condición

{ | sueldo\_A de tipo entero | }

### 1.3.3. Acción del algoritmo

Dar el aumento del empleado segun el sueldo del mes.

### 1.3.4. Post condición

{ | sueldo\_fin tipo real | }

## 2. Diseño

### 2.1. Descripción del algoritmo

#### Primera fase

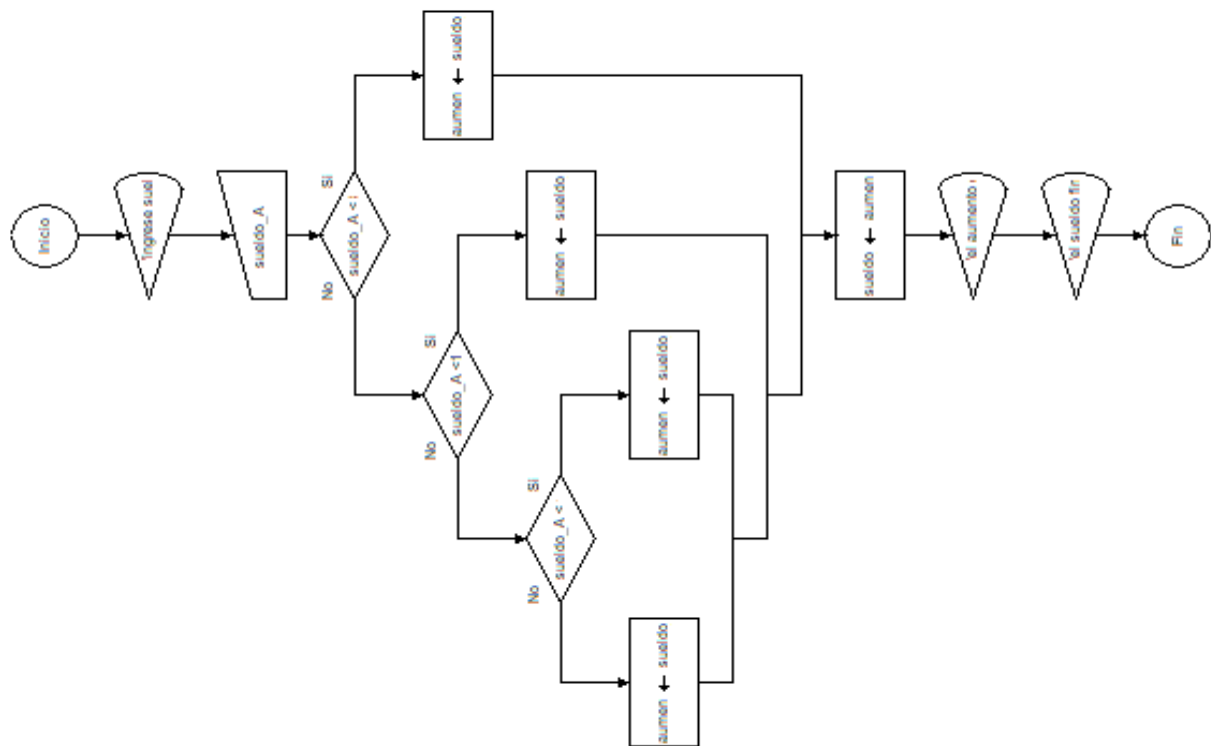
Inicio {determiner aumento del empleado }

{Declaración e inicialización de variables}

{Leer sueldo\_A}

{Determinar aumento }

{Mostrar la variable sueldo\_fin}



```

1  Proceso sin_titulo
2      definir sueldo_A como entero;
3      definir sueldo_fin , aumento como real;
4
5      Escribir "ingrese el sueldo:" ;
6      leer sueldo_A;
7
8      si (sueldo_A ≥ 0 ) y ( sueldo_A ≤ 800 ) entonces
9          .....
10         aumento ← sueldo_A * 0.2;
11     sino
12         si (sueldo_A ≥ 801 ) y ( sueldo_A ≤ 1000 ) Entonces
13             .....
14             aumento ← sueldo_A * 0.1;
15         sino
16             si (sueldo_A ≥ 1001) y ( sueldo_A ≤ 1500 ) Entonces
17                 .....
18                 aumento ← sueldo_A * 0.05;
19             sino
20                 si (sueldo_A > 1501 ) Entonces
21                     .....
22                     aumento ← sueldo_A * 0 ;
23                 FinSi
24             FinSi
25         FinSi
26     FinSi
27
28     sueldo_fin ← sueldo_A + aumento;
29
30     escribir "el aumento es:" ,aumento;
31     Escribir "el sueldo mas aumento es: " , sueldo_fin;
32
33 FinProceso

```

## “Año de la unidad, la paz y el desarrollo”

```
1  #include <iostream>
2
3
4  using namespace std;
5
6  int main() {
7
8      int sueldo_A;
9      float sueldo_fin , aumento;
10
11      cout<<"ingrese sueldo: ";
12      cin>>sueldo_A;
13
14      if ((sueldo_A >= 0 ) && ( sueldo_A <= 800 )){
15          aumento = sueldo_A * 0.2;
16      }
17      else{
18          if ((sueldo_A >= 801 ) && ( sueldo_A <= 1000 )){
19              aumento = sueldo_A * 0.1;
20          }
21          else {
22              if ((sueldo_A >= 1001 ) && ( sueldo_A <= 1500 )){
23                  aumento = sueldo_A * 0.05;
24              }
25              else {
26                  if (sueldo_A > 1500 ){
27                      aumento = sueldo_A * 0.0;
28                  }
29              }
30          }
31      }
32
33      sueldo_fin = sueldo_A + aumento;
34
35      cout<< "el aumento es:"<<endl<<aumento<<endl;
36      cout<< "el sueldo mas el aumento es: "<<endl<<sueldo_fin;
37
38      return 0;
39  }
```

## “Año de la unidad, la paz y el desarrollo”

- 2) La comisión sobre las ventas realizadas por un vendedor se calcula según las siguientes condiciones:

Venta realizada	% de comisión
0 a 200	0
Mas de 200 a 1000	10
Mas de 1000 a 2000	15
Mas de 2000 a 3000	20
Mas de 3000 a 4000	25
Mas de 4000	30

Desarrolle una aplicación que calcule las comisiones del vendedor.

Entrada	Proceso	Salida	
Sueldo_A		comision	Monto _comision
150		0%	0
620		10%	62
1250		15%	187.5
2800		20%	560
3600		25%	900
5200		30%	1560

### 1.2. Modelo

$$\text{Monto\_comision} = \begin{cases} \text{comision} = 0.0 & , ( \text{venta} > 0 ) \text{ y } ( \text{venta} < 200 ) \\ \text{comision} = 0.1 & , ( \text{venta} > 201 ) \text{ y } ( \text{venta} < 1000 ) \\ \text{comision} = 0.15 & , ( \text{venta} > 1001 ) \text{ y } ( \text{venta} < 2000 ) \\ \text{comision} = 0.20 & , ( \text{venta} > 2000 ) \text{ y } ( \text{venta} < 3000 ) \\ \text{comision} = 0.25 & , ( \text{venta} > 3000 ) \text{ y } ( \text{venta} < 4000 ) \\ \text{comision} = 0.30 & , ( \text{venta} > 4000 ) \end{cases}$$

### 1.3. Especificación del algoritmo

#### 1.3.1. Diccionario de variables

##### Variables de entrada

venta {representa las ventas del mes del vendedor }; tipo Entero

##### Variables de proceso

comision {representa la comision segun las ventas del vendedor }tipo real.

##### Variables de salida

Monto\_comision {representa la comision a recibir del vendedor }; tipo real.

#### 1.3.2. Pre condición

{ | venta de tipo entero | }

#### 1.3.3. Acción del algoritmo

Determiner la comision del vendedor .

#### 1.3.4. Post condición

{ | monto\_comision tipo real | }

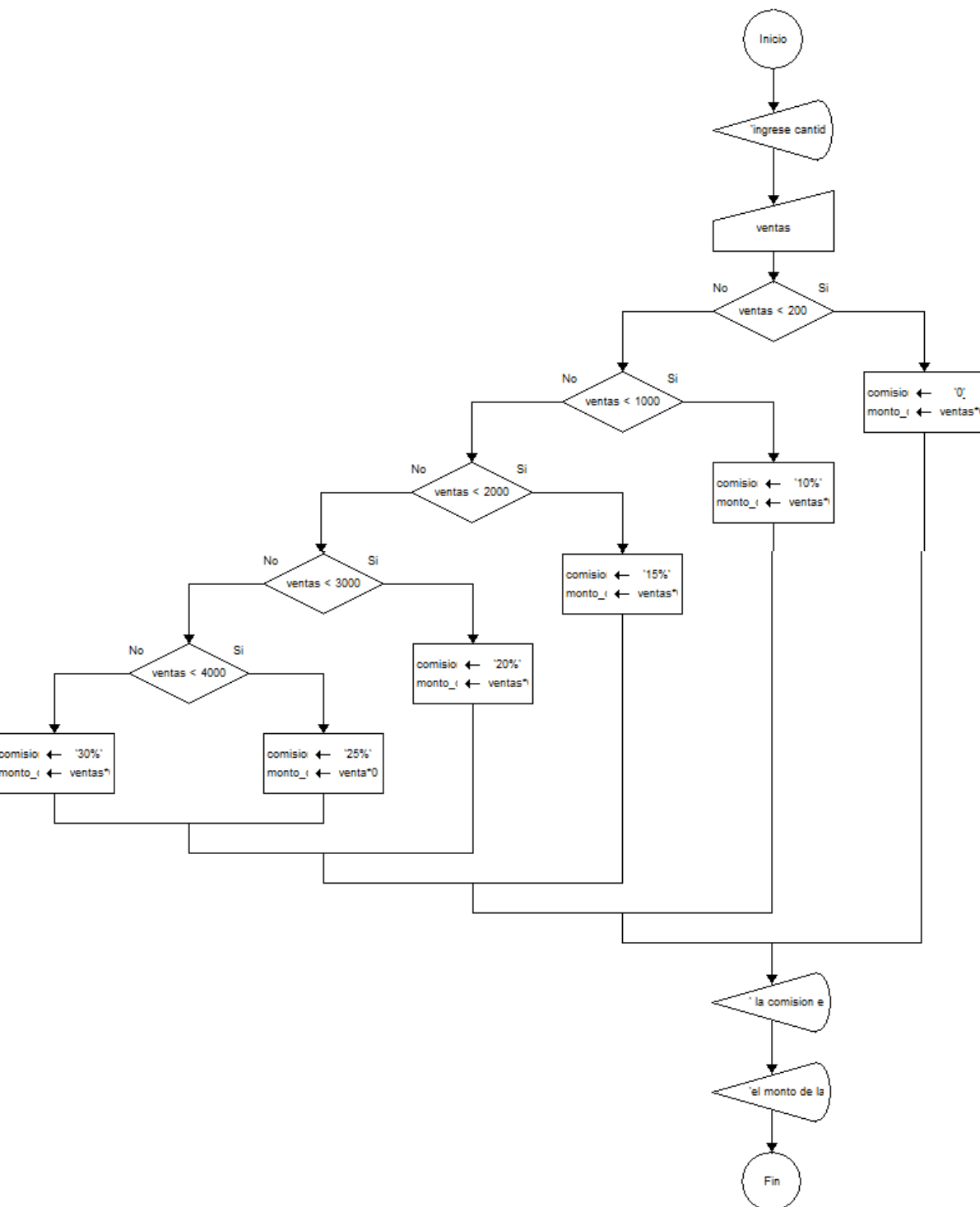
# “Año de la unidad, la paz y el desarrollo”

## 2. Diseño

### 2.1. Descripción del algoritmo

#### Primera fase

Inicio {determiner la comision del empleado }  
    {Declaración e inicialización de variables}  
    {Leer ventas}  
    {Determinar comision }  
    {Mostrar la variable monto\_comision }



## “Año de la unidad, la paz y el desarrollo”

```
1  Proceso sin_titulo
2      definir ventas como entero ;
3      definir comision Como Caracter;
4      definir monto_comicion como real;
5
6      Escribir "ingrese numero de ventas :";
7      leer ventas;
8
9      si (ventas ≥ 0 ) y ( ventas < 200) Entonces
10         comision ← "0%";
11         monto_comicion ← ventas;
12     sino
13         si (ventas ≥ 201 ) y ( ventas ≤ 1000) Entonces
14             comision ← "10%";
15             monto_comicion ← ventas*0.10;
16         sino
17             si (ventas ≥ 1001 ) y ( ventas ≤ 2000) Entonces
18                 comision ← "15%";
19                 monto_comicion ← ventas* 0.15;
20             sino
21                 si (ventas ≥ 2001 ) y ( ventas ≤ 3000) Entonces
22                     comision ← "20%";
23                     monto_comicion ← ventas* 0.2;
24                 sino
25                     si (ventas ≥ 3001 ) y ( ventas ≤ 4000) Entonces
26                         comision ← "25%";
27                         monto_comicion ← ventas * 0.25;
28                     sino
29                         comision ← "30%";
30                         monto_comicion ← ventas * 0.3;
31                     FinSi
32                 FinSi
33             FinSi
34         FinSi
35     FinSi
36
37     FinSi
38
39     FinSi
40
41     FinSi
42
43     Escribir "la comision es de el:",comision ;
44     Escribir "el monto de la comision es:", monto_comicion;
45
46 FinProceso
```

```

1  #include <iostream>
2
3  using namespace std;
4
5  int main (){
6
7
8      int ventas ;
9      string comision;
10     float monto_comision;
11
12     cout<<"ingrese numero de ventas: ";
13     cin>>ventas;
14
15     if ((ventas >=0 ) && ( ventas < 200)){
16         comision = "0%";
17         monto_comision = ventas*0.0;
18     }
19     else{
20         if ((ventas >=201 ) && ( ventas <= 1000)){
21             comision = "10%";
22             monto_comision= ventas * 0.1;
23         }
24         else{
25             if((ventas >=1001) && ( ventas <= 2000)){
26                 comision = "15%";
27                 monto_comision= ventas * 0.15;
28             }
29             else{
30                 if((ventas >=2001 ) && ( ventas <= 3000)){
31                     comision = "20%";
32                     monto_comision= ventas * 0.2;
33                 }
34                 else{
35                     if ((ventas >=3001 ) && ( ventas <= 4000)){
36                         comision = "25%";
37                         monto_comision= ventas * 0.25;
38                     }
39                     else{
40                         if (ventas > 4000){
41                             comision = "30%";
42                             monto_comision= ventas * 0.3;
43                         }
44                     }
45                 }
46             }
47         }
48     }
49     cout<<"la comision es de: "<<comision<<endl;
50     cout<<"el monto de la comision es de: "<<monto_comision;
51
52     return 0;
53 }

```



# “Año de la unidad, la paz y el desarrollo”

3) Sin usar estructuras repetitivas, lea 5 números y determine el mayor.

Entrada	Proceso	Salida
5 numeros		Num_mayor
12,13,16,18,9		16
1,12,8,10,11		12

## 1.2. Modelo

$$\text{Num\_mayor} = \left\{ \begin{array}{ll} \text{Num\_1} & (n1 > n2) \text{ y } (n1 > n3) \text{ y } (n1 > n4) \text{ y } (n1 > n5) \\ \text{Num\_2} & (n2 > n1) \text{ y } (n2 > n3) \text{ y } (n2 > n4) \text{ y } (n2 > n5) \\ \text{Num\_3} & (n3 > n1) \text{ y } (n3 > n2) \text{ y } (n3 > n4) \text{ y } (n3 > n5) \\ \text{Num\_4} & (n4 > n1) \text{ y } (n4 > n2) \text{ y } (n4 > n3) \text{ y } (n4 > n5) \\ \text{Num\_5} & (n5 > n1) \text{ y } (n5 > n2) \text{ y } (n5 > n3) \text{ y } (n5 > n4) \end{array} \right.$$

## 1.3. Especificación del algoritmo

### 1.3.1. Diccionario de variables

#### Variables de entrada

Num\_X(1,2,3,4,5) {representa el dato a ingresar }; tipo Entero

#### Variables de proceso

#### Variables de salida

Num\_mayor {guarda el dato con el numero mayor }; tipo entero.

### 1.3.2. Pre condición

{ | num\_X como real | }

### 1.3.3. Acción del algoritmo

Dar el numero mayor de los 5 ingresados

### 1.3.4. Post condición

{ | num\_mayor definido como entero | }

## 2. Diseño

### 2.1. Descripción del algoritmo

#### Primera fase

Inicio {dare 1 numero mayor }

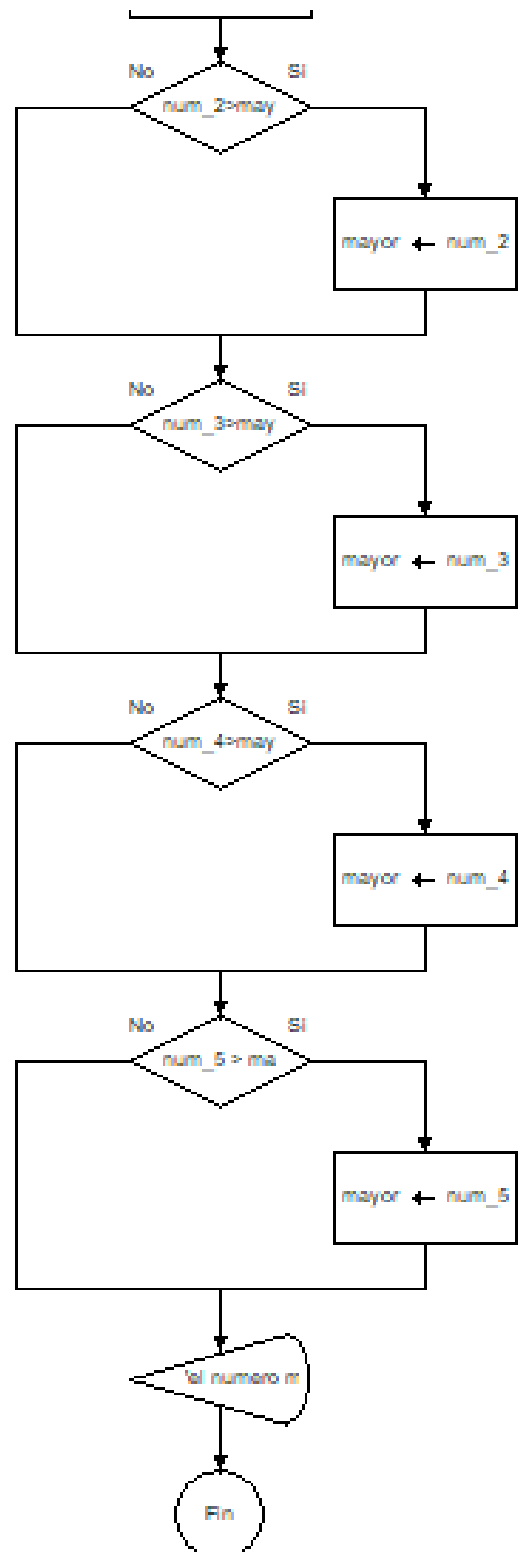
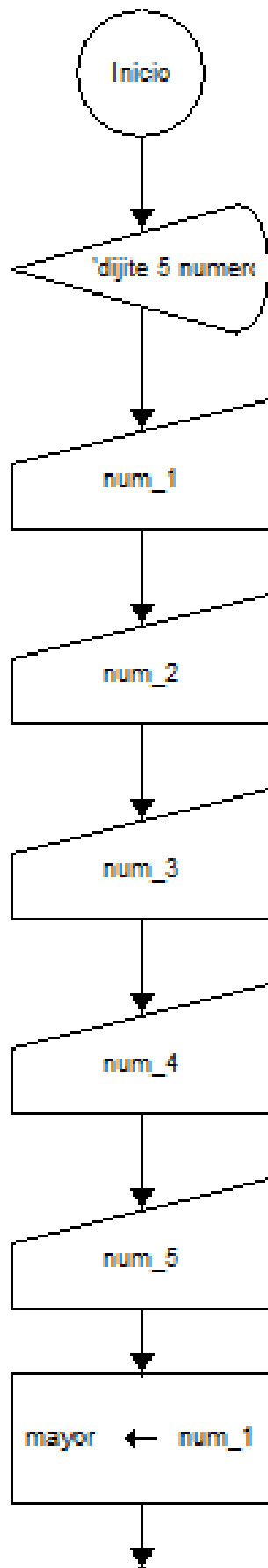
{Declaración e inicialización de variables}

{Leer num\_1,num\_2,num\_3,num\_4,num\_5}

{determiner el mayor }

{Mostrar el num\_mayor}

## “Año de la unidad, la paz y el desarrollo”



## “Año de la unidad, la paz y el desarrollo”

```
1  Proceso sin_titulo
2      Definir num_1 , num_2 , num_3 , num_4 , num_5 como entero;
3      definir num_mayor Como Entero;
4
5      escribir "dijite 5 numeros";
6      leer num_1;
7      leer num_2;
8      leer num_3;
9      leer num_4;
10     leer num_5;
11
12     si (num_1 > num_2 ) y (num_1 > num_3 ) y (num_1 > num_3 ) y (num_1 > num_5 ) Entonces
13         num_mayor ← num_1;
14     sino
15         si (num_2 > num_1 ) y (num_2 > num_3 ) y (num_2 > num_3 ) y (num_2 > num_5 ) Entonces
16             num_mayor ← num_2;
17         sino
18             si (num_3 > num_1 ) y (num_3 > num_2 ) y (num_3 > num_4 ) y (num_3 > num_5 ) Entonces
19                 num_mayor ← num_3;
20             sino
21                 si (num_4 > num_1 ) y (num_4 > num_2 ) y (num_4 > num_3 ) y (num_4 > num_5 ) Entonces
22                     num_mayor ← num_4;
23                 sino
24                     si (num_5 > num_1 ) y (num_5 > num_3 ) y (num_5 > num_3 ) y (num_5 > num_2 ) Entonces
25                         num_mayor ← num_5;
26                     FinSi
27                 FinSi
28             FinSi
29         FinSi
30     FinSi
31
32     Escribir "el numero mayor es: ", num_mayor;
33
34
35 finProceso
```

## “Año de la unidad, la paz y el desarrollo”

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main ()
6  {
7      int num_1,num_2,num_3,num_4,num_5;
8      int num_mayor;
9
10     cout<<"dijite 5 numero enteros: "<<endl;
11     cin>>num_1;
12     cin>>num_2;
13     cin>>num_3;
14     cin>>num_4;
15     cin>>num_5;
16
17     if ((num_1 > num_2 ) && (num_1 > num_3 ) && (num_1 > num_4 ) && (num_1 > num_5 )){
18         num_mayor = num_1;
19     }
20     else{
21         if ((num_2 > num_1 ) && (num_2 > num_3 ) && (num_2 > num_4 ) && (num_2 > num_5 )){
22             num_mayor = num_2;
23         }
24         else{
25             if ((num_3 > num_1 ) && (num_3 > num_2 ) && (num_3 > num_4 ) && (num_3 > num_5 )){
26                 num_mayor = num_3;
27             }
28             else{
29                 if ((num_4 > num_1 ) && (num_4 > num_2 ) && (num_4 > num_3 ) && (num_4 > num_5 )){
30                     num_mayor = num_4;
31                 }
32                 else{
33                     if ((num_5 > num_1 ) && (num_5 > num_2 ) && (num_5 > num_3 ) && (num_5 > num_4 )){
34                         num_mayor = num_5;
35                     }
36                 }
37             }
38         }
39     }
40     cout<<"el numero mayor es: "<<num_mayor;
41
42     return 0;
43 }
```

## “Año de la unidad, la paz y el desarrollo”

4) En una universidad tienen como política considerar 3 notas en cada curso la nota de trabajo T, la nota de medio ciclo M y la nota de fin de ciclo F, cada una tiene un peso de 50%, 20% y 30% respectivamente. Un alumno es calificado según lo siguiente:

Bueno	Si su promedio esta entre 16 y 20
Regular	Si su promedio esta entre 11 y 15
Malo	Si su promedio esta entre 6 y 10
Pésimo	Si su promedio esta entre 0 y 5

Escriba un programa que lea 3 notas de un alumno y en un curso y diga cómo ha sido catalogado el alumno.

Entrada			Proceso	Salida	
Nota_T	Nota_M	Nota_F		promedio	rendimiento
17	12	19	$17*0.5+12*0.2+19*0.3$	16.6	bueno
12	17	8	$12*0.5+17*0.2+8*0.3$	11.8	Regular
6	9	5	$6*0.5+9*0.2+5*0.3$	6.3	malo
4	3	1	$4*0.5+3*0.2+1*0.3$	2.9	Pesimo

### 1.2. Modelo

Rendimiento =	<div style="font-size: 4em; vertical-align: middle; margin-right: 10px;">{</div>	bueno	Si (promedio>16) y (promedio<20)
		regular	Si (promedio>11) y (promedio<15)
		malo	Si (promedio>06) y (promedio<10)
		pesimo	Si (promedio>00) y (promedio<05)

### 1.3. Especificación del algoritmo

#### 1.3.1. Diccionario de variables

##### Variables de entrada

Nota\_T {representa la nota de trabajo}: tipo Entero

Nota\_M {representa la nota de mitad de semestre}: tipo Entero

Nota\_F {representa la nota de fin de semestre}: tipo Entero

##### Variables de proceso

Promedio {representa el promedio final segun el valor de cada nota }tipo real.

##### Variables de salida

rendimiento {representa el rendimiento del estudiante }: tipo caracter.

#### 1.3.2. Pre condición

{ | nota\_T, nota\_M, nota\_F como entero | }

#### 1.3.3. Acción del algoritmo

determiner el promedio y rendimiento del estudiante.

#### 1.3.4. Post condición

{ | dar el promedio tipo real | }

{ | dar el rendimiento tipo caracter | }

# “Año de la unidad, la paz y el desarrollo”

## 2. Diseño

### 2.1. Descripción del algoritmo

#### Primera fase

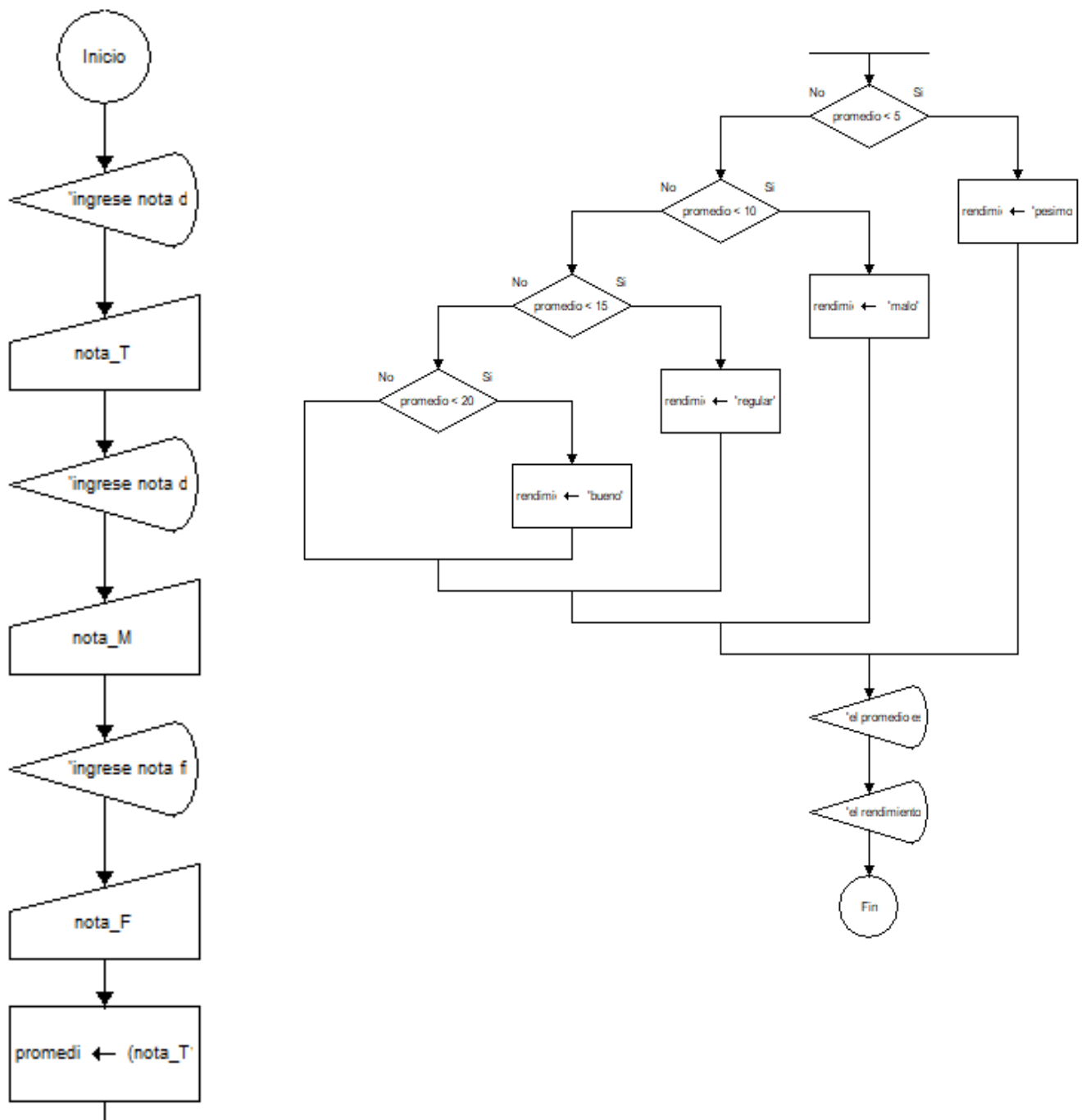
Inicio {determiner el rendimiento del alumno segun sus notas }

{Declaración e inicialización de variables}

{Leer nota\_T,nota\_M,nota\_F}

{Determinar promedio }

{Mostrar rendimiento }



## “Año de la unidad, la paz y el desarrollo”

```
1  Proceso sin_titulo
2      definir nota_T , nota_M , nota_F como entero ;
3      definir promedio como real;
4      definir rendimiento Como Caracter;
5
6      Escribir "dijite nota de trabajo: ";
7      leer nota_T;
8      Escribir "dijite nota de medio ciclo: ";
9      leer nota_M;
10     Escribir "dijite nota de fin de ciclo: ";
11     leer nota_F;
12
13     promedio ← (nota_T*0.50)+(nota_M*0.20)+(nota_F*0.30);
14
15     si (promedio ≥ 0 ) y (promedio < 5 ) Entonces
16     .....
17     .....
18     .....
19     .....
20     .....
21     .....
22     .....
23     .....
24     .....
25     .....
26     .....
27     .....
28     .....
29     .....
30     .....
31     .....
32
33     Escribir "el promedio del estudiante es de: ", promedio;
34     Escribir "el rendimiento del estudiante es ", rendimiento;
35
36
```

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main (){
6      int nota_T,nota_M,nota_F;
7      float promedio;
8      string rendimiento;
9
10     cout<< "dijite nota de trabajo: ";
11     cin>> nota_T;
12     cout<< "dijite nota de medio ciclo: ";
13     cin>> nota_M;
14     cout<< "dijite nota de fin de ciclo: ";
15     cin>> nota_F;
16
17     promedio = (nota_T*0.50)+(nota_M*0.20)+(nota_F*0.30);
18
19     if ((promedio >= 0 ) && (promedio < 5 )){
20         rendimiento ="pesimo";
21     }
22     else{
23         if((promedio >= 6 ) && (promedio < 10 )){
24             rendimiento="malo";
25         }
26         else{
27             if((promedio >= 11 ) && (promedio < 15 )){
28                 rendimiento="regular";
29             }
30             else{
31                 if((promedio >= 16 ) && (promedio < 20 )){
32                     rendimiento="bueno";
33                 }
34             }
35         }
36     }
37
38     cout<<"el promedio del estudiante es de: "<<promedio<<endl;
39     cout<<"el rendimiento del estudiante es: "<<rendimiento;
40
41     return 0;
42 }
```



# “Año de la unidad, la paz y el desarrollo”

5) Lea 2 enteros a,b y obtenga el valor numérico de la función:

$$f(x) \begin{cases} 2a+b & , \text{ si } a^2-b^2<0 \\ a^2-2b & , \text{ si } a^2-b^2=0 \\ a+b & , \text{ si } a^2-b^2>0 \end{cases}$$

1 analisis.-

1.1 entender el problema .-

Entrada		Proceso	Salida
Num_A	Num_B		Function
5	4		9
8	8		48
5	3		8

1.2. **Modelo**

$$\text{Function} \begin{cases} 2a+b & , \text{ si } a^2-b^2<0 \\ a^2-2b & , \text{ si } a^2-b^2=0 \\ a+b & , \text{ si } a^2-b^2>0 \end{cases}$$

1.3. **Especificación del algoritmo**

1.3.1. **Diccionario de variables**

**Variables de entrada**

Num\_A , num\_B {representan a los datos a ingresar }: tipo Entero

**Variables de proceso**

C, {una variable donde se guardara la ecuacion inicial }tipo real.

**Variables de salida**

Function {da como resultado la ecuacion segun la condicion en vase a la ecuacion inicial }: tipo real.

1.3.2. **Pre condición**

{ | num\_A , num\_B | }

1.3.3. **Acción del algoritmo**

Dar la function final en condicion a la ecuacion inicial ;

1.3.4. **Post condición**

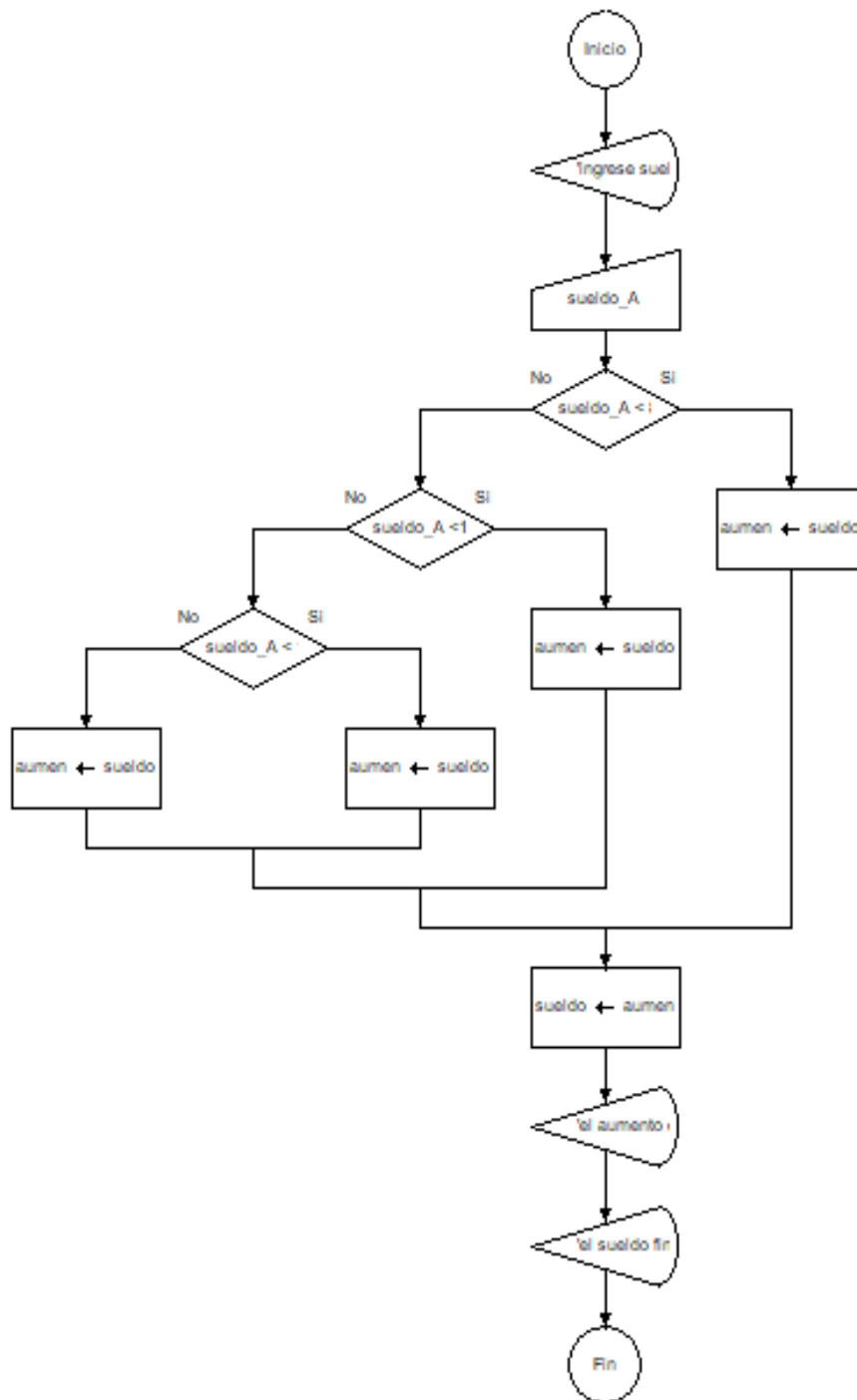
{ | function tipo real | }

## 2. Diseño

### 2.1. Descripción del algoritmo

#### Primera fase

Inicio {determiner aumento del empleado }  
    {Declaración e inicialización de variables}  
    {Leer sueldo\_A}  
    {Determinar aumento }  
    {Mostrar la variable sueldo\_fin}



```
1  Proceso sin_titulo
2      definir sueldo_A como entero;
3      definir sueldo_fin , aumento como real;
4
5      Escribir "ingrese el sueldo:" ;
6      leer sueldo_A;
7
8      si (sueldo_A ≥ 0 ) y ( sueldo_A ≤ 800 ) entonces
9      :      aumento ← sueldo_A * 0.2;
10     sino
11     :      si (sueldo_A ≥ 801 ) y ( sueldo_A ≤ 1000 ) Entonces
12     :      :      aumento ← sueldo_A * 0.1;
13     :      sino
14     :      :      si (sueldo_A ≥ 1001) y ( sueldo_A ≤ 1500 ) Entonces
15     :      :      :      aumento ← sueldo_A * 0.05;
16     :      :      sino
17     :      :      :      si (sueldo_A > 1501 ) Entonces
18     :      :      :      :      aumento ← sueldo_A * 0 ;
19     :      :      :      FinSi
20     :      :      FinSi
21     :      FinSi
22     FinSi
23
24     sueldo_fin ← sueldo_A + aumento;
25
26     escribir "el aumento es:" ,aumento;
27     Escribir "el sueldo mas aumento es: " , sueldo_fin;
28
29 FinProceso
30
```

```
1  #include <iostream>
2  #include <math.h>
3  using namespace std;
4
5  int main (){
6      int a , b,c,d,m,f;
7      int x;
8
9      cout<<"dijite dos numero 1 : "<<endl;
10     cin>>a;
11     cout<<"dijite numero 2: "<<endl;
12     cin>>b;
13
14     c = pow(a,2);
15     d = pow(b,2);
16
17     f = (c-d);
18
19     if ( f < 0 ){
20         x = 2*a - b;
21     }
22     else{
23         if (f == 0){
24             x = pow(a,2) - 2*b;
25         }
26         else{
27             if (f > 0){
28                 x = a + b ;
29             }
30         }
31     }
32
33     cout<<"el valor de la funcion es: "<<x;
34
35     return 0;
36 }
```