

PREUNIVERSITARIO

Práctico 2: Estructuras Condicionales

Soluciones propuestas

- Ejercicio 1

Escriba un programa que solicite la edad del usuario. Si el usuario es mayor de edad, deberá mostrar un mensaje en pantalla que lo indique.

```
Algoritmo Ejercicio1
    // Lo primero que haremos será definir una
variable, que contenga
    // la edad ingresada por el usuario
    definir edadUsuario como entero
    // Luego pasaremos a imprimir por consola la
pregunta
    escribir "¿Cuántos años tiene?"
    // Leemos la edad
    leer edadUsuario
    // Ahora crearemos un condicional que valide
laEje edad ingresada
    si edadUsuario<18 Entonces
        // Diremos que si es más chica, nos muestre
en la consola
        // que somos menores de edad y si no se
cumple esa condicion
        // que somos mayores
        escribir "¡Sos menor de edad!"
    SiNo
        escribir "¡Sos mayor de edad!"
    FinSi
FinAlgoritmo
```

- Ejercicio 2

Escriba un programa que solicite al usuario su sueldo actual y el sueldo mínimo vigente. Luego, el programa debe comparar ambos valores e informar al usuario si su sueldo es menor que el sueldo mínimo o si es superior.

Algoritmo Ejercicio2

```
// Empezamos declarando dos variable de tipo entero
que contendran el sueldo

// del usuario y el sueldo minimo.

definir sueldo Como Entero

Definir sueldoMin Como Entero

// Ahora preguntaremos por consola cuales es el sueldo
y de cuanto es el minimo.

escribir "¿De cuánto es su sueldo?"

// Leemos el valor

leer sueldo

escribir "¿De cuánto es el sueldo minimo actual?"

leer sueldoMin

// Terminaremos el ejercicio haciendo un condicional
que compare esas dos variables

// y nos imprima por consola el resultado

si sueldo<sueldoMin entonces

    escribir "Su sueldo es menor al sueldo minimo"

siNo

    escribir "Usted ganas más que el sueldo minimo"

finSi

finAlgoritmo
```

- Ejercicio 3

Realice un programa que pida un número al usuario. Si el número es mayor que 100, deberá mostrar el mensaje "Es Mayor"; en caso contrario, mostrará "Es Menor".

Algoritmo Ejercicio3

```
// Declaramos la variable que guardara el numero a
evaluar

definir num como entero
```

```
// Preguntamos cual es el numero a comparar  
  
escribir "Ingrese el numero a evaluar"  
  
// Leemos el número que ingreso el usuario  
  
leer num  
  
// Ahora haremos un condicional que evalúe si el  
número es  
  
// menor o mayor a 100 y nos diga el resultado  
  
si num<100 entonces  
  
    escribir "El número es menor a 100"  
  
SiNo  
  
    escribir "El número es mayor a 100"  
  
finSi  
  
finAlgoritmo
```

- **Ejercicio 4**

Escriba un programa que permita ingresar solo los caracteres 'S' y 'N'. Si el usuario ingresa uno de estos caracteres, el programa deberá mostrar en pantalla el mensaje "CORRECTO"; en caso contrario, deberá imprimir "INCORRECTO".

```
Algoritmo Ejercicio4  
    // Para empezar declararemos 3 variables, dos  
    // serán nuestras constantes S y N y la otra  
    // la usaremos para guardar la letra ingresada  
    // por el usuario  
    Definir letra Como Caracter  
    Definir s Como Caracter  
    Definir n como caracter  
    // Le asignaremos a nuestra variable S y N sus  
    // respectivos valores  
    s ="s"  
    n ="n"  
    // Le diremos por consola al usuario que ingrese  
    // una letra, luego leeremos el dato  
    escribir "Ingrese una letra"  
    leer letra
```

```
//Convertimos la letra ingresada a miniculas para
evitar un error si el usuario ingresa una mayuscula
letra = Minusculas(letra)

// Evaluaremos si la letra ingresada por el
usuario es igual a S o N y imprimiremos el resultado
si ((letra == s) O (letra == n)) entonces
    escribir "CORRECTO"
siNo
    escribir "INCORRECTO"
finSi

FinAlgoritmo
```

- Ejercicio 5

Realice un programa que solicite un número e indique si este es par o impar, mostrando un mensaje en pantalla que informe si el número es par o impar. Nota: investigar la función mod de PSeInt.

```
Algoritmo Ejercicio5
    // Empezaremos declarando una variable que
    contendra dentro el valor ingresado
    definir num como entero
    // Luego por consola le diremos al usuario que
    ingrese un numero para evaluarlo
    escribir "Ingrese un numero para evaluar si es
    par o no"
    // Leeremos el numero ingresado
    leer num
    // Y ahora con un condicional evaluaremos el
    numero ingresado e imprimiremos el resultado
    si (num mod 2 == 0) entonces
        escribir "El numero ingresado es par"
    siNo
        escribir "El número es impar"
    FinSi
finAlgoritmo
```

- Ejercicio 6

Escriba un programa que solicite tres calificaciones y determine si el alumno aprueba o reprueba el curso, sabiendo que se aprueba con un promedio igual o superior a 70.

```
Algoritmo Ejercicio6
```

```
// Empezaremos declarando tres variables que
seran correspondientes a las notas a promediar,
// luego declararemos otra variable que sera la
constante que contendra el promedio minimo para
aprobar

definir nota1 Como Entero
definir nota2 Como Entero
definir nota3 Como Entero
definir promedio como real

// Ahora preguntaremos y leeremos las notas por
consola
escribir "Ingrese la primera nota"
leer nota1
escribir "Ingrese la segunda nota"
leer nota2
escribir "Ingrese la tercera nota"
leer nota3

// Guardamos en nuestra variable promedio el
promedio de las tres notas
promedio = (nota1+nota2+nota3)/3

// Ya teniendo el promedio pasaremos a mostrar
los resultados a travez de un condicional
si (promedio <= 7) entonces
    escribir "Usted esta desaprobado"
siNo
    escribir "Usted está aprobado"
finSi

finAlgoritmo
```

- Ejercicio 7

Escriba un programa que permita introducir solo frases o palabras de 6 caracteres. Si el usuario ingresa una frase o palabra de 6 caracteres, el programa mostrará en pantalla el mensaje "CORRECTO"; en caso contrario, imprimirá "INCORRECTO". Nota: investigar la función Longitud() de PSeInt.

```
Algoritmo longitud_palabra
//Definimos la variables palabra
Definir palabra Como Caracter

//Pedimos al usuario que ingrese una palabra
```

```
    Escribir "Ingrese una palabra que tenga una
longitud de 6 caracteres: "
    Leer palabra

    //Comprobamos que la longitud de la palabra sea
IGUAL a 6.
    si(Longitud(palabra) == 6) Entonces
        //Si es igual a 6 se mostrará un mensaje de
"CORRECTO"
        Escribir "CORRECTO"
    SiNo
        //En el caso de que no lo sea se imprimirá
un mensaje de "INCORRECTO"
        Escribir "INCORRECTO"
    FinSi

FinAlgoritmo
```

- Ejercicio 8

Realice un programa que solicite una frase o palabra. Si la longitud es de 4 caracteres, el programa añadirá un signo de exclamación al final; si es diferente, añadirá un signo de interrogación. Luego, el programa mostrará la frase final. Nota: investigar las funciones Longitud() y Concatenar() de PSeInt.

```
Algoritmo concatenar_palabra
    //Definimos la variables palabra
    Definir palabra Como Caracter

    //Pedimos al usuario que ingrese una palabra
    Escribir "Dime una palabra!"
    leer palabra

    //Comprobamos que la longitud de la palabra sea
IGUAL a 4.
    si(Longitud(palabra) == 4) Entonces
        //si la frase o palabra es de 4 caracteres
de largo, se le sumará un signo de exclamación al
final
        Escribir Concatenar(palabra,"!")
    SiNo
        //Si no se le sumará un signo de
interrogación al final
        Escribir Concatenar(palabra,"?")
    FinSi
```

FinAlgoritmo

- Ejercicio 9

Una tienda ofrece un descuento de 500 pesos durante los meses de septiembre, octubre y noviembre. Escriba un programa que solicite el mes y el importe de compra, y calcule el total a cobrar aplicando el descuento en caso de corresponder.

Algoritmo ejercicio9

```
//Definimos las variables a utilizar
definir mes Como Caracter
definir importe Como Real
definir importeFinal Como Real

//Pedimos al usuario que ingrese un mes y un
importe
Escribir "Ingresar Mes en letras: "
Leer mes
Escribir "Ingresar importe de la compra: "
Leer importe

//Convertimos el texto que contenga la variable
me a minusculas, y lo volvemos a almacenar en la
variale
//para evitar errores al comparar luego en el
condicional
mes = Minusculas(mes)

//Si el mes es igual a 'septiembre', 'octubre' o
'noviembre' se aplicará un descuento
si(mes == "septiembre" O mes == "octubre" O mes
== "noviembre")Entonces
    importeFinal = importe - 500
    Escribir "Se aplicó $500 de descuento"
SiNo
    importeFinal = importe
    Escribir "No se aplicaron descuentos"
FinSi

Escribir "El importe final es de: ", importeFinal

FinAlgoritmo
```

- Ejercicio 10

Escriba un programa que solicite dos números enteros y determine si ambos son pares o impares, mostrando en pantalla el mensaje "Ambos números son pares" solo si cumplen la condición. En caso contrario, mostrará "Los números no son pares, o uno de ellos no es par".

Nota: investigar la función mod de PSeInt.

Algoritmo ejercicio10

```
//Definimos las variables
definir numUno como entero
Definir numDos Como Entero

//Le pedimos al usuario que ingrese ambos números
por consola
    escribir "Escribir el primer número: "
    leer numUno
    escribir "Escribir el segundo número: "
    leer numDos

//Un número es par si el módulo (resto) de un
número dividido entre 2 da 0.
    si((numUno MOD 2 == 0) Y (numDos MOD 2 ==
0))Entonces
        //Si ambos números son par devolvemos "Ambos
números son pares"
        Escribir "Ambos números son pares"
    SiNo
        //Si no devolvemos "Los números no son
pares, o uno de ellos no es par"
        Escribir "Los números no son pares, o uno de
ellos no es par"
    FinSi

FinAlgoritmo
```

- Ejercicio 11

Escriba un programa que solicite tres calificaciones y valide si estas están entre 1 y 10. Si cumplen con estos valores, se asignará verdadero a una variable lógica; en caso contrario, se asignará falso. Al final, el programa deberá indicar si las tres calificaciones son válidas usando la variable lógica.

Algoritmo notas_correctas

```
//Definimos las variables
```



```
Definir notas Como Entero

Definir i Como Entero

Definir notaIngresada Como Entero

//La variable notas la definimos como un ARREGLO y en
la línea siguiente le asignamos una longitud

Dimension notas[3]

Definir esNotaCorrecta Como Logico


//Mediante una iteración PARA le pedimos al usuario
que ingrese una nota que luego será asignada al ARREGLO
notas

Para i = 1 Hasta 3 Con Paso 1 Hacer

    Escribir "Dime la nota n° ", i

    leer notaIngresada

    notas[i-1] = notaIngresada

Fin Para

//Comprobamos que todas las notas sean mayor o igual a
1 y menor o igual a 10

si((notas[0] >= 1 Y notas[0] <= 10) Y (notas[1] >= 1 Y
notas[1] <= 10) Y (notas[2] >= 1 Y notas[2] <= 10))entonces

    //Si es así la variable esNotaCorrecta será
verdadera

    esNotaCorrecta = Verdadero

SiNo

    esNotaCorrecta = Falso

FinSi

si(esNotaCorrecta == Verdadero)Entonces

    Escribir "Las 3 notas son correctas"

SiNo
```

```
Escribir "Una o más de las 3 notas es incorrecta"
```

```
FinSi
```

```
FinAlgoritmo
```

- Ejercicio 12

Escriba un programa que solicite una frase o palabra y valide si la primera letra es una 'A'. Si es así, deberá imprimir en pantalla "CORRECTO"; en caso contrario, mostrará "INCORRECTO".

Nota: investigar la función Subcadena() de PSeInt.

```
Algoritmo frase_a
```

```
//Definimos la variable  
definir frase Como Caracter
```

```
//Pedimos al usuario que ingrese por consola una  
frase
```

```
Escribir "Escribe una frase o una palabra: "  
leer frase
```

```
//Convertimos la frase a minusculas para que no  
de error al comparlo en el condicional  
frase = Minusculas(frase)
```

```
//Comprobamos si la frase que introdujo el  
usuario comienza con una letra A  
si(SubCadena(frase,0,0) == "a")Entonces  
    Escribir "Correcto, la frase comienza con A"  
SiNo  
    Escribir "INCORRECTO, la frase no comienza  
con A"  
FinSi
```

```
FinAlgoritmo
```

- Ejercicio 13

Partiendo del ejercicio anterior, realice un programa que solicite una frase o palabra y valide si la primera letra coincide con la última. Si es así, mostrará "CORRECTO"; en caso contrario, imprimirá "INCORRECTO".

```
Algoritmo Ejercicio13
```

```
//Se pedirá una frase o palabra y se validara si  
la primera letra
```

```
//de la frase es igual a la última letra de la
frase. Se deberá
//de imprimir un mensaje por pantalla que diga
//"CORRECTO", en caso contrario, se deberá
imprimir "INCORRECTO".

//Declaración de variables
Definir frase Como Caracter
Definir primer_letra Como Caracter
Definir ultima_letra Como Caracter

Escribir "Ingrese una palabra o frase: "
Leer frase //Asignamos el ingreso por teclado a
la variable 'frase'

//Convertimos toda la frase a minuscula para una
correcta comparacion
frase = Minusculas(frase)

//Tomamos la primera letra de 'frase' y la
asignamos a la variable 'primer_letra'
primer_letra = Subcadena(frase, 0, 0)
//Tomamos la última letra de 'frase' y la
asignamos a la variable 'ultima_letra'
ultima_letra = Subcadena(frase, Longitud(frase)-
1, Longitud(frase)-1)

//Comparamos las variables 'primer_letra' y
'ultima_letra'
Si primer_letra == ultima_letra Entonces
    Escribir "CORRECTO"; //Acción si son iguales
SiNo
    Escribir "INCORRECTO"; //Acción si no son
iguales
FinSi

FinAlgoritmo
```

- Ejercicio 14

La empresa "Te llevo a todos lados" alquila autos por hora. Si el cliente devuelve el auto en menos de 2 horas, deberá abonar \$400 y el combustible será gratuito. Si supera las 2 horas, se ingresarán la cantidad de litros de combustible consumidos y las horas de uso. El programa deberá calcular el total a pagar considerando \$40 por litro y \$5,20 por minuto.

Algoritmo Ejercicio14

```
//Declaración de variables
Definir tiempo_alquiler Como Real
Definir litros Como Real
Definir minutos Como Real
Definir total Como Real
total = 0
minutos = 0

Escribir "Ingrese el tiempo de alquiler del auto
en horas: "
Leer tiempo_alquiler //Asignamos el tiempo
ingresado a la variable 'tiempo_alquiler'

Si tiempo_alquiler <= 2 Entonces
    total = 400 //Accion si el tiempo no supera
    las 2 horas
SiNo //Si el tiempo supera las 2 horas pedimos el
ingreso de los litros de nafta consumidos
    Escribir "Ingrese la cantidad de litros
consumidos:"
    Leer litros //Asignamos la cantida ingrasada
a la variable 'litros'
    minutos = tiempo_alquiler*60 //Convertimos
la cantidad de horas ingresadas a minutos
    total = (litros*40) + (minutos*5.2)
//Calculamos el total a pagar por el cliente
FinSi

//Mostramos el total a pagar por pantalla
Escribir "El total a pagar por el cliente es de
$",total

FinAlgoritmo
```

- Ejercicio 15

Escriba un programa que solicite un número del 1 al 7 y muestre el día de la semana correspondiente. Considere que el número 1 corresponde a "Lunes", y así sucesivamente.

Algoritmo Ejercicio15

```
//Declaración de variables
Definir num Como Entero
```

```
//Pedimos al usuario que ingrese un numero
Escribir "Ingrese un número entre 1 y 7:"
Leer num //Asignamos el número ingresado a la
variable 'num'

Escribir "El número ingresado corresponde al
día:";

Segun num Hacer //Según el número ingresado se
mostrará el día correspondiente por pantalla
1:
    Escribir "Lunes"

2:
    Escribir "Martes"

3:
    Escribir "Miércoles"

4:
    Escribir "Jueves"

5:
    Escribir "Viernes"

6:
    Escribir "Sábado"

7:
    Escribir "Domingo"
De Otro Modo:
    Escribir "El número ingresado está
fuera del rango requerido." //Mensaje que se mostrará
en caso que el número no corresponda a ningún día
Fin Segun

FinAlgoritmo
```

- Ejercicio 16

Escriba un programa que simule un menú de opciones para realizar operaciones aritméticas básicas (suma, resta, multiplicación y división) entre dos números enteros. El usuario debe especificar la operación con la primera letra (S para suma, R para resta, M para multiplicación, D para división).

Algoritmo Ejercicio16

```
//Declaracion de variables
Definir num1 Como Entero
```

```
Definir num2 Como Entero
Definir operacion Como Caracter
Definir total Como Real

//Pedimos el ingreso de los números y asignamos a
las variables 'num1' y 'num2'
Escribir "Ingrese el primer número a operar: "
Leer num1
Escribir "Ingrese el segundo número a operar: "
Leer num2

//Pedimos el ingreso de la inicial de la
operacion
Escribir "Ingrese la inicial de la operación a
realizar:"
Escribir "s para suma,"
Escribir "r para resta,"
Escribir "m para multiplicacion y"
Escribir "d para division."
Leer operacion //Asignamos la letra ingresada a
la variable 'operacion'
operacion = Minusculas(operacion) //Convertimos
la variable 'operacion' a minusculas para trabajar con
ella

//Segun la letra ingresada se realizará la
operacion indicada, de no corresponder a ninguna
operacion, se muestra un mensaje por pantalla
segun operacion Hacer
    "s":
        total = num1 + num2
    "r":
        total = num1 - num2
    "m":
        total = num1 * num2
    "d":
        total = num1 / num2

De Otro Modo:
    Escribir "La operacion ingresada es
invalida"

FinSegun
```

```
//Mostramos el resultado obtenido por pantalla  
Escribir "El resultado de la operación elegida  
entre ",num1," y ",num2," es ",total
```

```
FinAlgoritmo
```

- Ejercicio 17

Escriba un programa que solicite una fecha (día, mes, año) y valide si es correcta. Si es válida, deberá imprimirla con el nombre del mes; por ejemplo, "1 de febrero de 2006". En caso contrario, mostrará un mensaje de error.

Algoritmo Ejercicio17

```
//Declaracion de variables  
Definir dia Como Entero  
Definir mes Como Entero  
Definir anio Como Entero  
Definir mes_letra Como Caracter  
Definir aux Como Logico  
  
aux = Falso  
  
//Pedimos al usuario que ingrese un numero para  
el dia, uno para el mes y uno para el año y los  
asignamos a las variables correspondientes  
Escribir "Ingrese un número para el día:"  
Leer dia  
  
Escribir "Ingrese un número para el mes:";  
Leer mes  
  
Escribir "Ingrese un número para el año:";  
Leer anio  
  
//Analizamos la validez de la fecha segun la  
cantidad de dias del mes  
Si (mes==1 O mes ==3 O mes==5 O mes==7 O mes==10  
O mes==12) Y (dia>0 Y dia<32) Entonces //Condicion  
meses de 31 dias  
    aux = Verdadero  
  
Sino Si mes==2 Y (((anio % 4 == 0) Y (anio % 100  
<> 0) O (anio % 400 == 0)) Y dia<30) O dia<29)
```

```
Entonces //Condicion febrero, incluyendo si el año es
bisiesto
    aux = Verdadero

    Sino Si (mes==4 O mes==6 O mes==8 O mes==9 O
mes==11) Y (dia>0 Y dia<31) Entonces //Condición meses
de 30 días
        aux = Verdadero
    FinSi
FinSi
FinSi

//Escribimos el mes en letras
Segun mes Hacer
1:
    mes_letra = "Enero"
2:
    mes_letra = "Febrero"
3:
    mes_letra = "Marzo"
4:
    mes_letra = "Abril"
5:
    mes_letra = "Mayo"
6:
    mes_letra = "Junio"
7:
    mes_letra = "Julio"
8:
    mes_letra = "Agosto"
9:
    mes_letra = "Septiembre"
10:
    mes_letra = "Octubre"
11:
    mes_letra = "Noviembre"
12:
    mes_letra = "Diciembre"
Fin Segun

//Si la fecha es válida la mostramos por
pantalla, de lo contrario mostramos mensaje de fecha
invalida
Si aux Entonces
```



```
        Escribir dia, " de ",mes_letra," de ",anio
SiNo
        Escribir "Fecha inexistente"
FinSi
```

```
FinAlgoritmo
```

- Ejercicio 18

Escriba un programa que, dado un número entero, indique si es par o impar. Si el valor es 0, mostrará "El número no es par ni impar". Nota: investigar la función mod de PSeInt.

Algoritmo Ejercicio18

```
//Definicion de variables
Definir num Como Entero

//Pedimos al usuario que ingrese un numero y lo
asignamos a la variable 'num'
Escribir "Ingrese un número:"
Leer num

//Analizamos el numero ingresado
Si num == 0 Entonces
    Escribir "El número no es par ni impar"
//Accion si el número ingresado es 0
Sino
    Si (num MOD 2) == 0 Entonces //Comprobamos
    si el número es par, analizando el valor del resto de
    la division entre el número y 2
        Escribir "El número es par"
    SiNo
        Escribir "El número es impar"
    FinSi
FinSi

FinAlgoritmo
```

- Ejercicio 19

Escriba un programa que, dado un año, determine si es bisiesto. Un año es bisiesto si es divisible por 4, pero no por 100, excepto si también es divisible por 400. Nota: investigar la función mod de PSeInt.

Algoritmo anio_bisiesto

```
// Declaración de variables
Definir anio Como Entero
Definir esAnioBisiesto Como Logico

// Pedido de datos al "usuatio"
Escribir "Ingrese un año, sin comas o puntos"
Leer anio

esAnioBisiesto = Falso

// Es año bisiesto, si es divisible por 4 y el
resto da cero ademas no es divisible por 100.
Si ( anio mod 4 == 0 Y anio mod 100 <> 0)
Entonces
    esAnioBisiesto = Verdadero
FinSi

// Es año bisiesto, si es divisible por 100 y
ademas es divisible por 400
Si ( anio mod 100 == 0 Y anio mod 400 == 0)
    esAnioBisiesto = Verdadero
FinSi

// Mostramos por consola si el numero ingresado
es un año bisiesto
si ( esAnioBisiesto ) Entonces
    Escribir "El año ingresado es bisiesto"
SiNo
    Escribir "El año ingresado no es bisiesto"
FinSi
FinAlgoritmo
```

- **Ejercicio 20**

Escriba un programa para calcular el grado de eficiencia de un operario según estas condiciones:

- Producir menos de 200 tornillos defectuosos.
- Producir más de 10,000 tornillos sin defectos.

Se asignará el grado de eficiencia según estas reglas:

- Grado 5: si no cumple ninguna de las condiciones.
- Grado 6: si solo cumple la primera condición.

- Grado 7: si solo cumple la segunda.
- Grado 8: si cumple ambas condiciones. Nota: pruebe cada paso del ejercicio de forma progresiva.

Algoritmo grado_eficiencia

```
// Declaro variables

Definir tornillosDefectuosos Como Entero

Definir tornillosSinDefectos Como Entero

Definir eficiencia Como Entero

//Solicitud de datos

Escribir "Ingrese la cantidad de tornillo defectuosos"

Leer tornillosDefectuosos

Escribir "Ingrese la cantidad de tornillo sin
defectos"

Leer tornillosSinDefectos

// Verificar las condiciones para asignar un grado de
eficiencia

// Primera comprobación : la cantidad de tornillos
defectuoso es "menor" a 200 Y la cantidad

// de tornillos sin defectos es "mayor" a 10000

Si ( tornillosDefectuosos < 200 Y tornillosSinDefectos
> 10000 ) Entonces

    eficiencia = 8

FinSi

// Segunda comprobación : la cantidad de tornillos
defectuoso es "menor" a 200 Y la cantidad

// de tornillos sin defectos es "menor o igual" a
10000

Si ( tornillosDefectuosos < 200 Y
tornillosSinDefectos <= 10000 ) Entonces

    eficiencia = 7
```

```
FinSi

// Tercera comprobación : la cantidad de tornillos
defectuoso es "mayor o igual" a 200 Y la cantidad

// de tornillos sin defectos es "mayor" a 10000

Si ( tornillosDefectuosos >= 200 Y
tornillosSinDefectos > 10000 ) Entonces

    eficiencia = 6

FinSi

// Primera comprobación : la cantidad de tornillos
defectuoso es "mayor o igual" a 200 Y la cantidad

// de tornillos sin defectos es "menor o igual" a
10000

Si (tornillosDefectuosos >= 200 Y tornillosSinDefectos
<= 10000 ) Entonces

    eficiencia = 5

FinSi

// Mostramos por consola el grado de eficiencia del
operario

Escribir "El grado de eficiencia del operario es : ",
eficiencia

FinAlgoritmo
```

- Ejercicio 21

Una empresa cuenta con personal bajo distintas modalidades de contratación. Escriba un programa que calcule el sueldo semanal en base a tres tipos de sueldo:

- a) Comisión: el sueldo es el 40% del monto total de ventas de la semana.
- b) Salario fijo + comisión: se ingresan las horas trabajadas (máximo 40) y el total de ventas, calculando el 25% de comisión sobre las ventas.
- c) Salario fijo: se ingresan las horas trabajadas, y si exceden las 40, las horas extra se pagan con un 50% de aumento.

```
Algoritmo empresa_contratacion
//Declaro las variables a utilizar
Definir salario Como Real
```

```
Definir horasTrabajadas Como Entero
Definir montoPorHora Como Real
Definir horasExtra Como Entero
Definir montoPorHoraExtra Como Real
Definir totalVentas Como Real
Definir comision Como Real
Definir respuesta Como Caracter
Definir
montoTotalConsola, horasTrabajoConsola, montoPorHoraConsola Como Real

//Obtener respuesta del usuario para calcular el salario
Escribir "Ingrese [ A ] para calcular el Salario por Comisión de venta"
Escribir "Ingrese [ B ] para calcular el Salario + Comisión"
Escribir "Ingrese [ C ] para calcular el Salario"
Leer respuesta

// Usamos estructura de control "Switch" o "Segun el caso" como alternativa a multiples "Si"

Segun respuesta Hacer
    "A" o "a":
        // Solicitamos nuevamente los datos al usuario
        Escribir "Ingrese el monto total de ventas "
        Leer montoTotalConsola

        // Asignamos los valores dados por consola a nuestras variables
        totalVentas = montoTotalConsola

        // Calculamos la comision que es el cuarenta por ciento ( 40% o 0.4 ) del total de ventas
        comision = totalVentas*0.4
        // Como la opcion elegida fue "A" el salario equivale al 40% de las ventas
```

```

    salario = comision

    // Mostramos por consola lo calculado
    Escribir "El salario por comisiones
fue del : $", salario

    "B" o "b":
        // Solicitamos nuevamente los datos al
usuario
        Escribir "Ingresa la cantidad de horas
trabajadas"
        Leer horasTrabajoConsola

        Escribir "Ingrese el monto por hora
trabajada"
        Leer montoPorHoraConsola

        Escribir "Ingrese el monto total de
ventas "
        Leer montoTotalConsola

    // Asignamos los valores dados por
consola a nuestras variables
    totalVentas = montoTotalConsola
    horasTrabajadas = horasTrabajoConsola
    montoPorHora = montoPorHoraConsola

    // Verificamos que el total de horas
trabajadas sea "menor o igual" a cuarenta ( 40 hs )
    Si ( horasTrabajadas <= 40) Entonces
        // No hizo más de '40 hs'
        //la comision por ventas es del
25%
        comision = totalVentas*0.25
        // Como la opción elegida fue "B"
el salario se calcula como :
        // "horas trabajadas" por "monto a
pagar por hora" más la comisión
        salario = horasTrabajadas *
montoPorHora
        salario = salario + comision
```

```
SiNo
    // Supera las '40 hs' de trabajo,
    segun el enunciado del ejercicio para la opcion "B",
    las horas extras
    // no se contemplan y se fija un
    máximo de 40 como referencia a '40 hs'
    comision =totalVentas*0.25
    salario = 40 * montoPorHora

    salario = salario + comision
FinSi

    // Mostramos por consola lo calculado
    Si ( horasTrabajadas <= 40) Entonces
        Escribir "El salario + comisiones
sin horas extras fue del : $", salario
    SiNo
        Escribir "El salario + comisiones
con horas extras fue del : $", salario
    FinSi

Caso "C" o "c":
    // Solicitamos nuevamente los datos al
usuario
    Escribir "Ingresa la cantidad de horas
trabajadas"
    Leer horasTrabajoConsola

    Escribir "Ingresa el monto por hora
trabajada"
    Leer montoPorHoraConsola

    // Asignamos los valores dados por
consola a nuestras variables
    horasTrabajadas = horasTrabajoConsola
    montoPorHora = montoPorHoraConsola

    // Verificamos que el total de horas
trabajadas sea "menor o igual" a cuarenta ( 40 hs )
    Si ( horasTrabajadas <= 40) Entonces
        // No hizo más de '40 hs'
        // Como la opción elegida fue "C"
el salario se calcula como :
```

```
// "horas trabajadas" por "monto a
pagar por hora"
montoPorHora = horasTrabajadas *
salario

SiNo
    // Supera las '40 hs' de trabajo,
    segun el enunciado del ejercicio para la opcion "C",
    las horas extras
    // Si se contemplan
    // Para calcular las horas extras
    a las horas trabajadas les restamos el valor de 40, ya
    que como dice el
    // enunciado del ejercicio el máximo
    permiso de horas es 40, el resto se considera "horas
    extra"
    horasExtra = horasTrabajadas - 40

    // las horas extras se pagan con
    una bonificación del 50% sobre el monto por hora
    montoPorHoraExtra = montoPorHora +
    montoPorHora*0.5

    // Finalmente calculamos el
    salario como las "horas trabajadas" por el "monto a
    cobra por hora"
    salario = ( 40 * montoPorHora )
    // más las "horas extras" por el
    "monto de bonificación" del 50%
    salario = salario + ( horasExtra *
    montoPorHoraExtra )
FinSi
    // mostramos por consola lo calculado
    Si ( horasTrabajadas <= 40) Entonces
        Escribir "El salario sin horas
        extras fue del: $", salario
    SiNo
        Escribir "El salario con horas
        extras fue del: $", salario
    FinSi
FinSegun
FinAlgoritmo
```