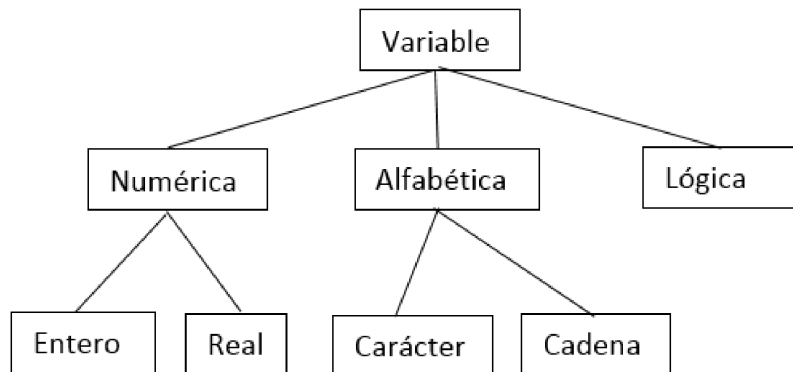


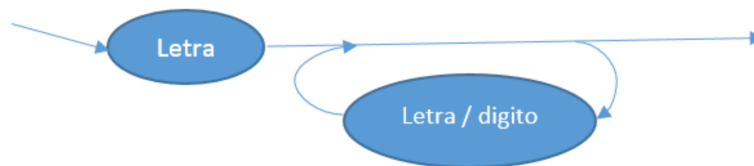
Introducción a la Programación Estructurada

Conceptos

Variable: Es un identificador que almacena un valor final. Una clasificación básica es la siguiente:



Se define una variable de la siguiente forma:



Se asigna valores numéricos, alfabéticos o lógicos a las variables. Por ejemplo:

$a \leftarrow 15$; $b \leftarrow \text{"hola"}$; $c \leftarrow 0$; $aux \leftarrow -1$; $con \leftarrow \text{false}$; $sum25 \leftarrow 3.4$; $pepe \leftarrow 23251$; $pi \leftarrow 3.1416$;

Las variables cambian durante la ejecución de un programa, las constantes no.

Operadores Aritméticos:

\wedge Exponente
 $*$ / Multiplicación - División
 $+$ - Suma - Resta

Operadores de Relación:

$<$ $>$ \leq \geq $=$ $<>$

Operadores lógicos

not
 and
 or

Expresión Aritmética: Conjunto de variables, constantes, valores finales y operadores aritméticos. El resultado que devuelve es un valor numérico.

Ejemplo: $a \leftarrow 2$; $b \leftarrow 3$;
 $a \wedge b + 5 - a * b \wedge a / 6 - 1$ El resultado es : 9

Expresión de Relación: Conjunto de Expresiones aritméticas y operadores de relación.

Ejemplo: $a \leftarrow 2$; $b \leftarrow 3$;
 $a \wedge b > b * a$ Verdad (true)

El resultado devuelve un valor lógico (falso o verdadero)

Expresión lógica: Conjunto de expresiones de relación y operadores lógicos.

Ejemplo: $a \leftarrow 2$; $b \leftarrow 3$;
 $a \wedge b > b * a$ and $b * a > a \wedge a$ Verdad (true)

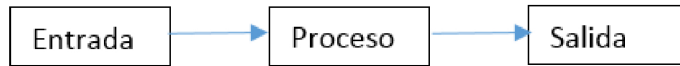
El resultado devuelve un valor lógico (falso o verdadero)

Programación Estructurada:

La programación estructurada responde a tres estructuras básica:

- Secuencial: un solo camino
- Condicional: caminos alternativos
- Repetitiva: caminos que se repiten

La estructura básica de procesamiento de datos se resume:



Una expresión aritmética se puede asignar a una variable de la siguiente forma:

$$b \leftarrow a * c;$$

Donde a y c son variables independientes y b es la variable dependiente.

Por lo tanto las variables independientes se convierten por lo general en variables de entrada y las variables dependientes en variables de salida. El proceso está representado por la expresión aritmética.

Ejercicio 1. Por lo que el algoritmo para esta expresión es:

INICIO

```

read( a, c );    //variables independientes
b ← a * c;      //expresión aritmética asignando a una variable
print (b);      //variable de salida
  
```

FIN

Si se conoce la expresión aritmética el algoritmo es inmediato. Pero hay muchos ejercicios de programación que no necesariamente, se conocen las expresiones aritméticas, por lo tanto, es necesario realizar un diseño para definir las.

Ejercicio 2. Dado un número de 5 dígitos, eliminar el dígito central

Ejemplo: Establecemos un numero cualquiera de 5 dígitos, en lo posible un buen ejemplo con dígitos diferentes para poder analizar y lograr descubrir las expresiones aritméticas

El número para este ejemplo es: 56238

Entrada	Proceso	Salida
56238	¿?	5638

El proceso se identificará en distintas variables de la siguiente forma

n	a	b	c
56238	56	38	5638

Para obtener 56 en a, se debe eliminar 3 dígitos, por lo tanto, se extrae la parte entera de dividir n entre 1000.

$$a \leftarrow || n / 1000 ||$$

Para obtener 38 en b, n se somete a módulo 100, el resto de dividir n entre 100

$$b \leftarrow n \% 100$$

Para obtener 5638, se debe multiplicar 100 a 56 y sumarle 38

$$c \leftarrow a * 100 + b$$

El algoritmo queda:

INICIO

```
read (n);
a ← || n / 1000 ||
b ← n % 100
c ← a * 100 + b
print (c);
```

FIN

Ejercicio 3. Dado un número de 6 dígitos, eliminar el dígito que representa el dígito de la decena
Ejemplo: Establecemos un numero cualquiera de 6 dígitos, en lo posible un buen ejemplo con dígitos diferentes para poder analizar y lograr descubrir las expresiones aritméticas

El número para este ejemplo es: 756483

Entrada	Proceso	Salida
756483	¿?	75643

El proceso se identificará en distintas variables de la siguiente forma

n	a	b	c
756483	7564	3	75643

Para obtener 3 en b, se debe sacar $b \leftarrow n \% 10$ y asignar a la variable b. Luego debemos eliminar 83 de n, y este nuevo número almacenarlo en a con la siguiente operación $a \leftarrow ||n/100||$. Posteriormente unimos a y b con una expresión aritmética, $c \leftarrow a * 10 + b$

El algoritmo queda:

INICIO

```
read (n);
b ← n % 10
a ← ||n/100||
c ← a * 10 + b
print (c);
```

FIN

Practica 1. INF111

(Debe estar ordenadamente resuelto en su cuaderno de prácticas, debe presentar para ingresar a clases a hrs 8:00 am, jueves 15/08/24)

Resolver el área de las siguientes figuras geométricas:

1. Trapecio
2. Pentágono regular
3. Octógono regular
4. Paralelogramo
5. Circunferencia
6. Dado un número de 5 dígitos intercambiar los dígitos extremos
7. Dado un número de 8 dígitos sumar de 2 en 2 dígitos
8. Dado un número de 7 dígitos eliminar el dígito central y reemplazar en los extremos.