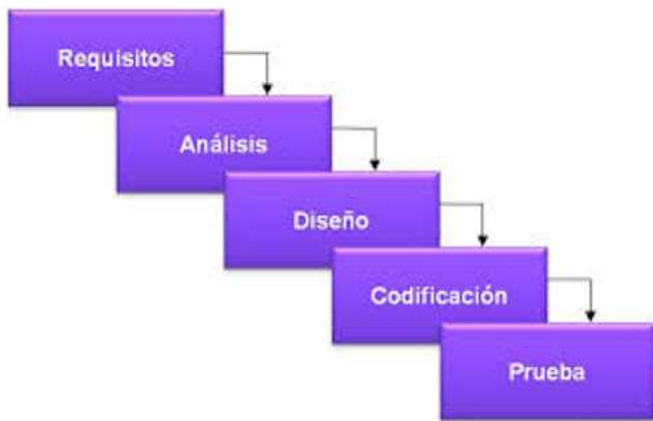


14/04/2023

Carpeta de Algoritmos

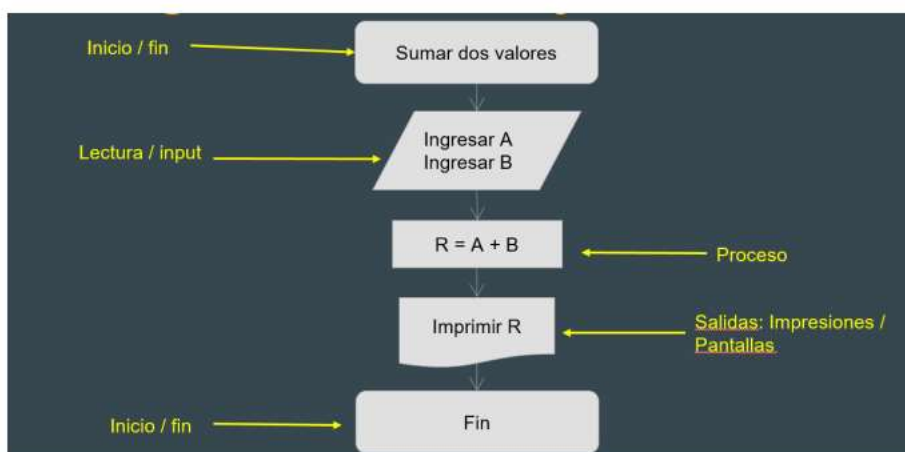









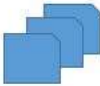

Un programa es una serie de algoritmos que llevan a un resultado específico.

Veamos lo que sucede en cada fase, con algo más de detalle:

- **Análisis de Requisitos y del Problema:** El problema se analiza teniendo presente la especificación de los requisitos dados por el cliente de la empresa o por otra persona que encarga el programa.
- **Diseño del algoritmo:** Una vez analizado el problema, se diseña una solución que conducirá a un algoritmo que resuelva el problema. Codificación. La solución se escribe en la sintaxis del lenguaje de alto nivel (por ejemplo, C) y se obtiene un programa
- **Codificación, verificación y depuración:** El programa se ejecuta, se comprueba rigurosamente y se elimina todos los errores (denominados “bugs”, en inglés) que puedan aparecer
- **Mantenimiento y Pruebas:** El programa se actualiza y modifica, cada vez que sea necesario, de modo que se cumplan todas las necesidades de cambio de sus usuarios.
- **Documentación:** Escritura de las diferentes fases del ciclo de vida del software, esencialmente el análisis, diseño y codificación, unidos a manuales de usuario y referencia, así como normas para el mantenimiento. Esta actividad debe realizarse siempre como parte del proceso, ya que la documentación es un complemento fundamental del software desarrollado.

Diagramas de flujo:



SIMBOLO	SIGNIFICADO
	Operación: Se usa para describir cualquier actividad. En el interior del rectángulo se escribe una breve descripción de la actividad.
	Límites del Proceso: Indica el inicio y el final de un proceso. En el interior del óvalo aparece la palabra inicio o fin.
	Punto de Decisión: Denota que en ese punto se toma una decisión. Los outputs salidas del diamante, son siempre dos y del tipo Si / No.
	Movimiento: Muestra el movimiento de un output entre distintos puntos de la organización.
	Conector: Señala que el output de ese proceso puede ser el input de otro (la letra indica el proceso de entrada)
	Dirección del flujo: Denota la dirección y el orden de los pasos del proceso
	Documento: Documento/registro.
	Listados: Listados / notas de trabajo acumulado, información referente a la actividad.
	Base de datos: Punto de archivo donde se retiene temporalmente la información, en espera que se cumplan otras condiciones para continuar el proceso. Puede llevar asociada una tarea de administración de almacenamiento.

Operaciones: Se utilizan + - * / ^ por ejemplo.

Conectores: Hay dos tipos de conectores;

- El círculo: que es un conector dentro de de la pagina.
- Una flecha: es un conector, pero esta vez fuera de la pagina.

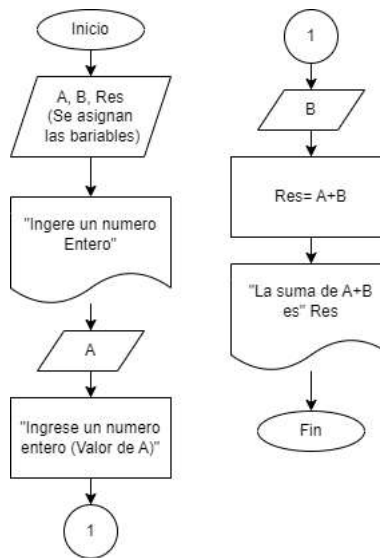
Variables:

- Son un espacio que se ocupa en la memoria.
- Estos tienen un tipo de dato en específico, Variables (Estas son las que cambian constantemente) y Constantes (Pueden ser nombres de empresas, el valor de pi, etc) estas son las variables que no cambian con el tiempo.
- Tienen su valor específico.
- Hábito.
- Ciclo de vida, esto quiere decir que tiene principio y final.

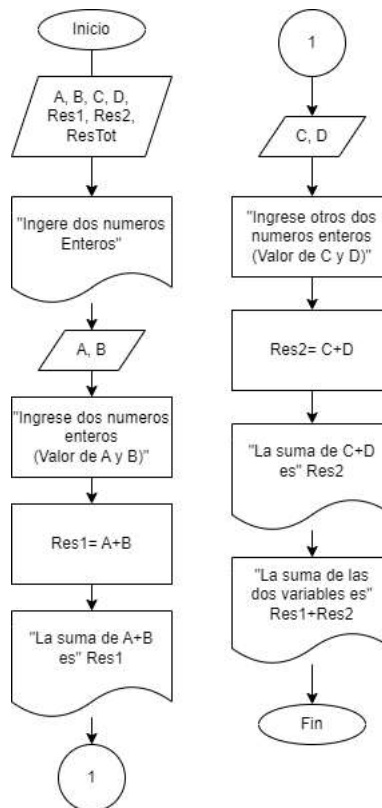
A) Ingresar dos números enteros, sumarlos, y mostrarlos en pantalla.

B) Ingrese 4 números enteros, sume de a 2 y muestre el resultado final.

A)



B)





Empezamos el programa por declarar las variables, estas en este caso van a ser 7:

- Variable 1
- Variable 2
- Variable 3
- Variable 4
- Resultado 1
- Resultado 2
- Resultado total

Luego con esto podemos empezar a calcular los resultados, si sumamos la variable 1 con la variable 2 obtenemos resultado 1.

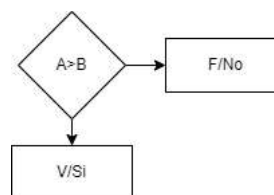
Seguido de esto podemos mostrar el resultado en pantalla, esto va a mostrar la suma de ambos números.

Si seguimos el programa podemos sumar la variable 3 y la variable 4, esto nos daría el resultado 2.

Por ultimo podemos sumar los dos resultados, esto seria sumar resultado 1 y resultado 2, que da como respuesta resultado total.

Con que controlo el flujo de datos?

Con condicionales y ciclos. (PREGUNTA DE PARCIAL)



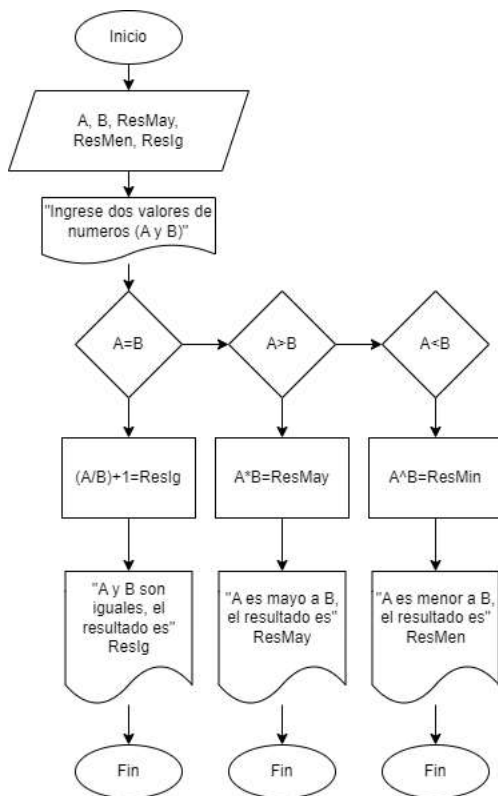
Para demostrar igualdades o desigualdades se usan los siguientes operadores:

- $>$ (Mayor que)
- $<$ (Menor que)
- \geq (Mayor o igual)
- \leq (Menor o igual)
- $==$ (Igualdad)
- $!=$ (No igual)

Ingresa 2 números enteros;

- Si el primero es mayor que el segundo, realizar el producto y mostrarlo.
- Si el primero es menor que el segundo, elevar el primero a la potencia con el valor del segundo.
- Si son iguales y el segundo es distinto de 0, realizar el cociente del primero dividido el segundo +1.

En todos los casos mostrar el resultado.



Ingrese la longitud de los 3 lados de un triangulo.

Comparar sus lados e indicar si el triangulo es, equilátero, isósceles o escalno.

21/04/2023

Pregunta de parcial:

Con que se controla el flujo de datos?

Con los condicionales y ciclos repetitivos

Ingrese una fecha e indique si es una fecha valida, y ademas si el año es bisiesto.

Febrero puede tener 28 o 29 días dependiendo si es bisiesto.

Los meses 4, 6, 9, 11 tienen 30 días.

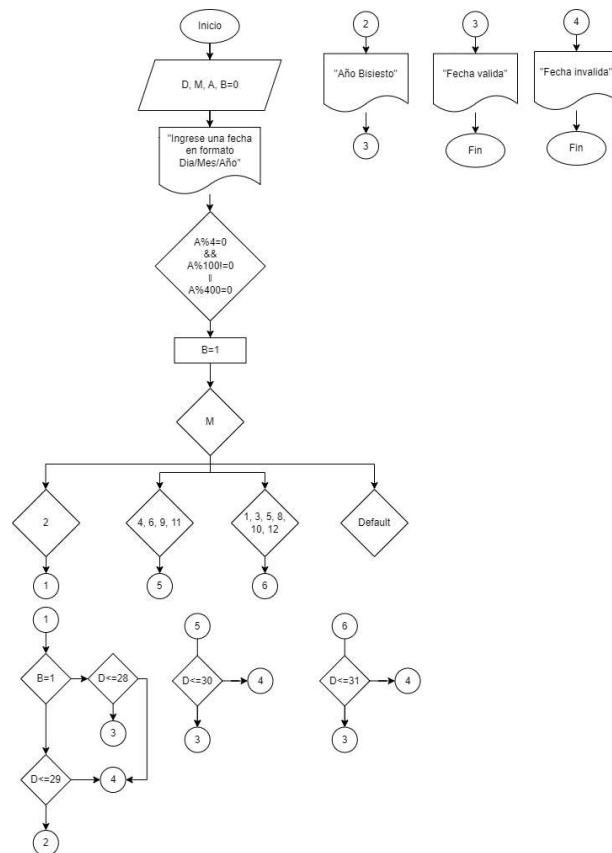
Los meses 1, 3, 5, 8, 10, 12 tienen 31 días.

Si el año es divisible por 4 y su resto es 0 ($\text{Mod} = 0$) y (AND, $\&\&$) dividido 100 su resto es distinto de 0 este es bisiesto.

Si el año se divide por 100 y su resto es 0 ($\text{Mod} = 0$) este es bisiesto.

Switch:

- Casos
- Casos
- Default



Preguntas de parcial

Variable (Puede ser constante o variable, eso quiere decir que puede cambiar o no):

Existen distintos tipos de variables, entre estas están **{Int; Float; Char; Double}**

Estas tienen las siguientes características:

- Tipo de dato.
- Ciclo de vida.
- Espacio de memoria.
- **Ámbito.**

El ámbito puede ser; Global; Función; Local.

- [Valor]
- Tienen nombre.

Ejemplos de variables constantes:

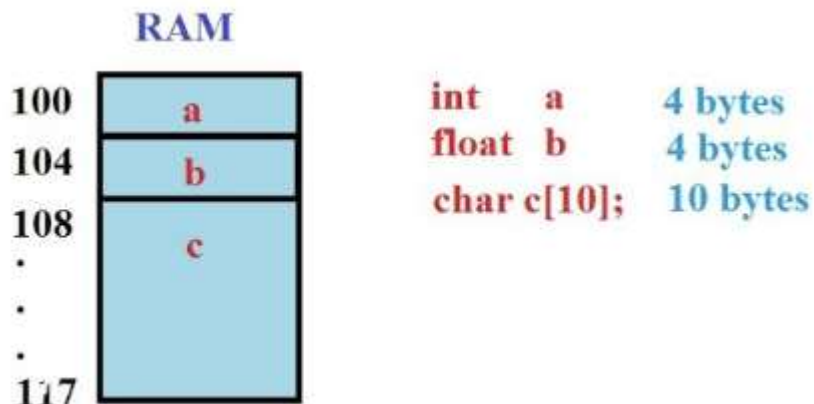
- El nombre de una empresa.
- EL Cuil o el Cuit.

(Estos son los datos que no van a cambiar a lo largo del programa)

#define empresa "UAI"

(La **#** significa que es una instrucción de preprocesador, lo primero que se ejecuta)

Espacio que ocupan en la memoria:



Teorema de la programación:

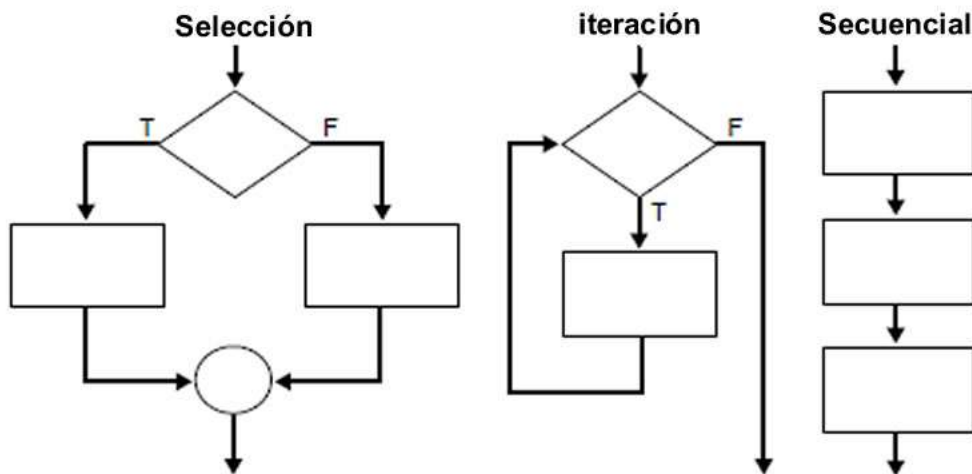
El teorema del programa estructurado es un resultado en la teoría de lenguajes de programación. Establece que toda función computable puede ser implementada en un lenguaje de programación que combine sólo tres estructuras lógicas. Esas tres formas específicamente son: Secuencia: ejecución de una instrucción tras otra.

Tipos de ciclos:

Exactas; “para” == “for”

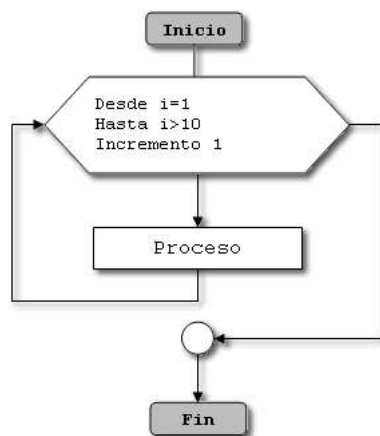
No exactas;

- “mientras” == “while”
- “haga mientras” == “do...while”



Antes de esto se define la variable que se va a usar dentro de la función:

```
int i;
```



i=1

(Entre el inicio y la comparación se tiene que definir la variable)

i<=10

i++

Este es un ejemplo del loop, lo que hace es incrementar el numero inicial “i” que en este caso vale 1, este lo incrementa de 1 en 1, hasta que “i” valga 11, cuando valga 11 este loop se terminara.

Se puede incrementar de a 1, de a 2, de a 3, o de a cuantos quieras, hasta en negativo.

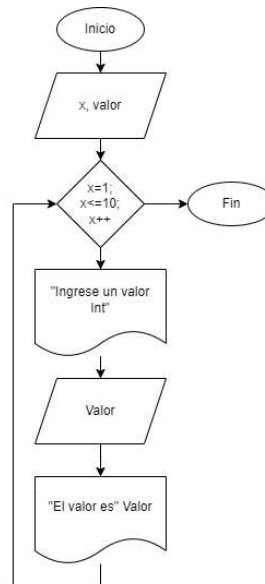
Ejemplo:

Para incrementar de a 1 se utiliza el ++

Para decrecer de a 1 se utiliza el --

Para incrementar o decrecer de a mas de 1 se usa lo siguiente:

- $x = x + 2$
- $x = x + 10$
- $x = x - 4$
- $x = x - 11$



```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int main(int argc, char *argv[]) {
```

```
    int control, suma = 0, valor, producto = 1;
```

```
    float promedio;
```

```
    for (control = 1; control <= 25; control++)
```

```
    {
```

```
        printf("\nIngrese el valor de un entero ");
```

```
        scanf("%d", &valor);
```

```
        suma += valor;
```

```
    }
```

```
    promedio+=(float)suma/25;
```

```
    printf("\nSuma %d y promedio %.2f", suma, promedio);
```

```
    return 0;
```

```
}
```

Ingresa cinco valores enteros, sumarlos, y multiplicarlos por el índice.

\\t es para hacer el tab

Al ejercicio 7 agregar, contar cuantos son pares y cuantos son impares.

Indicar que porcentaje representa la suma de los valores positivo, negativo, ceros, pares e impares sobre el total de los números cargados.

05/05/2023

Como controlo el flujo de datos

Secuencial (Top down)

Selección → if -- Switch

interacción →

- For (Exacto)
- While (Mientras)
- do... While

Los if no pueden existir si no tienen salida en la verdad, puede tener solo la salida de la verdad sin necesidad de salida en el falso, pero no viceversa.

Estructura del ciclo for

El for tiene 3 parámetros

for (____;____;____)

El primero es el inferior, desde donde empieza. (x=1)

El segundo es el superior, la cantidad de veces que se va a repetir el for. (x=n)

El tercero es el paso, cuantas veces lo voy a incrementar. (x++ o x+=2)

El while

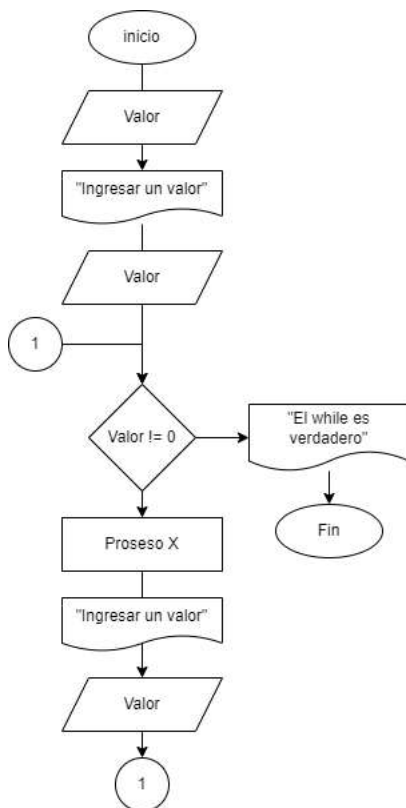
Los operadores que podemos encontrar pueden ser:

- == (Igual)
- != (Distinto)
- > (Mayor que)
- < (Menor que)
- >= (Mayor o igual que)
- <= (Menor o igual que)

While: este se va a repetir hasta que la condición se cumpla la instancia entre paréntesis.

- while (x<10)
- while (x!=0)
- while (x>100)
- while (x>100 && <25)

Para que el while se cumpla, antes del mismo, debe existir la variable que me permite entrar al ciclo.



El do while

Se pueden usar los mismos operadores que en el while.

La estructura es:

```
do {} while ();
```