



# Universidad Nacional Autónoma de México

Facultad de Ingeniería



## Estructura de datos y algoritmos 1

*Profesor(a): M.I. Marco Antonio Martínez Quintana.*

*Semestre 2021-2*

Nombre de la actividad

Actividad 1: Repaso de lo aprendido en fundamentos de programación

**Integrante:**

García Martínez Sammy Adolfo

Fecha de entrega Cd. Universitaria a 26 de Febrero de 2021.

## **REPASO DE FUNDAMENTOS DE PROGRAMACION.**

En este documentó realizare un pequeño resumen de todo lo que vi y aprendí en la materia de fundamentos de programación.

En esa materia iniciamos viendo la definición y lo que es un algoritmo, en el cual empezamos haciendo ejercicios de algoritmos de la vida diaria como el hacer un pastel los pasos que se lleva para realizar dicho pastel, preparar una torta,etc.

Después de eso ya pasamos a los algoritmos con las palabras reservadas, donde también ocupábamos el programa de pseint, hicimos muchos ejercicios como los siguientes:

### **Algoritmo primero**

```
Definir x,c Como Entero
Escribir 'dame 15 numeros y veras que pasara'
Mientras c<15 Hacer
    Escribir 'dame un numero'
    Leer x
    Si x=0 Entonces
        Escribir x,'es neustro'
    SiNo
        Si x<0 Entonces
            Escribir x,'es negativo'
        SiNo
            Escribir x,'es positivo'
    FinSi
    c <- c+1
FinSi
FinMientras
FinAlgoritmo
```

### **Algoritmo tercero**

```
Definir a,b,c,d,s,x1,x2 Como Entero
```

```

Escribir 'dame a,b,c'
Leer a,b,c
Si a!=0 Entonces
    d <- (b*b)-(4*a*c)
    Si d>=0 Entonces
        Si d=0 Entonces
            s <- -b/2*a
            Escribir s
        SiNo
            x1 <- -b+rc(b*b)/(2*a)
            x2 <- -b-rc(b*b)/(2*a)
            Escribir "x1 es ",x1,"x2 es",x2
        FinSi
    SiNo
        Escribir 'imaginario'
    FinSi
SiNo
    Escribir 'indefinido'
FinSi
FinAlgoritmo

```

En la parte de laboratorio , empezamos a ver la introducción a Linux, así como sus comandos para compilar y correr el programa , así como saber buscar en google las cosas y saber usar los comandos del buscador de google para graficar una ecuación o para buscar una imagen etc.

Después de eso lo más importante que vimos fue empezar a ver diagramas de flujo, el cual el objetivo era Elaborar diagramas de flujo que representen soluciones algorítmicas vistas como una serie de acciones que comprendan un proceso.

Donde vimos los puntos de cómo se tiene que componer cada diagrama de flujo las cuales son :

1. Todo diagrama de flujo debe tener un inicio y un fin
2. Las líneas utilizadas para indicar la dirección del flujo del diagrama deben ser rectas, verticales u horizontales, exclusivamente.
3. Todas las líneas utilizadas para indicar la dirección del flujo del diagrama deben estar conectadas a un símbolo.
4. El diagrama debe ser construido de arriba hacia abajo y de izquierda a derecha.
5. La notación utilizada en el diagrama de flujo debe ser independiente del lenguaje de programación en el que se va a codificar la solución
6. Se recomienda poner comentarios que expresen o ayuden a entender un bloque de símbolos.
7. Si la extensión de un diagrama de flujo ocupa más de una página, es necesario utilizar y numerar los símbolos adecuados.
8. A cada símbolo solo le puede llegar una línea de dirección de flujo.
9. Notación de camello. Para nombrar variables y nombres de funciones se debe hacer uso de la notación de camello.

Ya sabiendo esto me enseñaron el significado de cada una de las figuras que componen el diagrama de flujo las cuales son las siguientes:



Inicio y final del diagrama.



## Operaciones.



Datos de entrada.



Datos de salida.



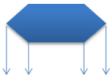
Dirección del flujo del diagrama.



Conector dentro de la misma página.



Conector para diferentes páginas.

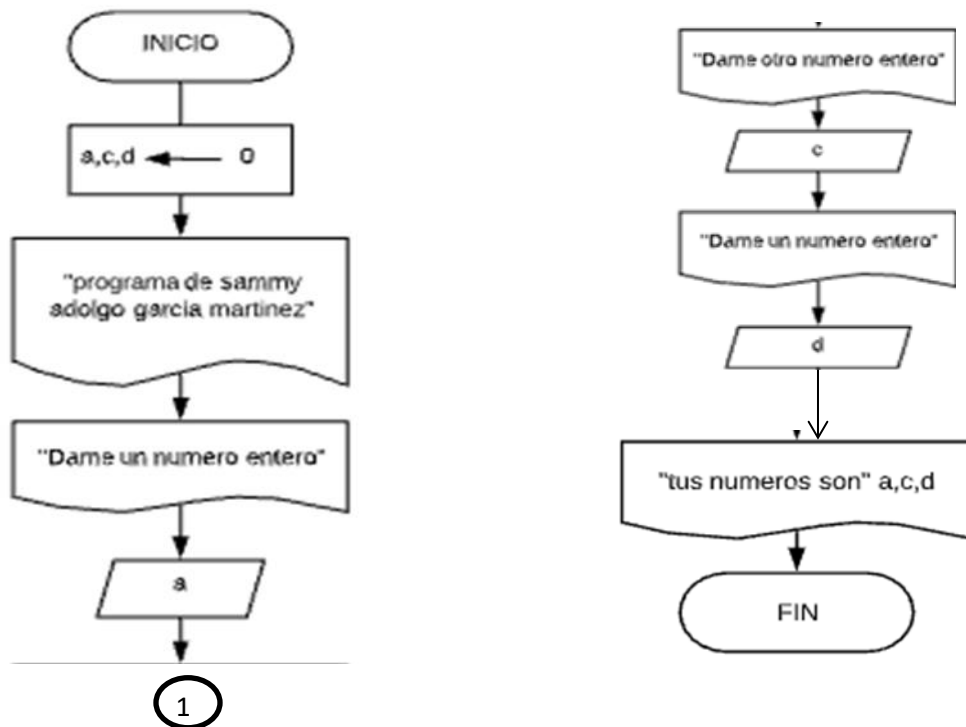


Decisión múltiple.

Después de saber todo eso supimos realizar diagramas de flujo como el siguiente ejemplo:

Leer 3 números enteros y mostrarlos.





Posteriormente ,pasamos a ver el tema de pseudocodigo donde el objetivo principal era elaborar pseudocódigos que representen soluciones algorítmicas empleando la sintaxis y semántica adecuadas.

Al igual que el diagrama de flujo vimos su estructura del Pseudocodigo el cual es el siguiente:

1. Alcance del programa: Todo pseudocódigo está limitado por las etiquetas de INICIO y FIN. Dentro de estas etiquetas se deben escribir todas las instrucciones del programa.
2. Palabras reservadas con mayúsculas: Todas las palabras propias del pseudocódigo deben de ser escritas en mayúsculas.

3. Sangría o tabulación: El pseudocódigo debe tener diversas alineaciones para que el código sea más fácil de entender y depurar.

4. Lectura / escritura: Para indicar lectura de datos se utiliza la etiqueta LEER. Para indicar escritura de datos se utiliza la etiqueta ESCRIBIR. La lectura de datos se realiza, por defecto, desde el teclado, que es la entrada estándar del sistema. La escritura de datos se realiza, por defecto, en la pantalla, que es la salida estándar del sistema.

Y después de eso realizamos ejercicios aquí un ejemplo que realizamos:

Leer 2 números y mostrar las cuatro operaciones fundamentales entre los dos, usando sólo una variable para mostrar los resultados Sin utilizar seleccionador múltiple.

Análisis

Entrada: números enteros

Salida: números enteros y letrero

Restricciones: verificar que sean diferentes de cero

INICIO

$x \leftarrow -0$     $b \leftarrow -0$     $c \leftarrow -0.0$

ESCRIBIR "Ingresa dos números que quieras sumar, restar, multiplicar y dividir."

LEER x,y

MIENTRAS  $y \neq 0$

    MOSTRAR "Ingrese un numero diferente a cero"

    LEER y

FIN MIENTRAS

```
c<- x+y
```

```
MOSTRAR "El resultado de la suma es:" c
```

```
c<- x-y
```

```
MOSTRAR "El resultado de la resta es:" c
```

```
c<- x*y
```

```
MOSTRAR "El resultado de la multiplicación es:" c
```

```
c<- x/y
```

```
MOSTRAR "El resultado de la división es:" c
```

FIN

Después de ver toda esta parte ,pasamos a la parte de la programación donde empezamos trabajando en el compilador de C y vimos sus características y como se trabajaba en el , vimos sus diferentes librerías entre las cuales usamos una de estas librerías que se denotan de la siguiente manera:

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

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

```
#include<math.h>
```

Entre otras, aprendí a declarar variables y depende de que tipo de variable se utilizaría se declaraba ya sea con int que es para enteros, float que es para flotantes,char para caracteres, y adentro de la programación nos enseñaron a utilizar el if,while, do while, for y switch.



En donde for ,while y do while se usan para los ciclos, if es una estructura de decisión y switch es un seleccionador,después de conocer e implementar estas técnicas pasamos a ver lo que es arreglos que esos se declaran entre corchete, por ejemplo:

```
Int arreglo[15];
```

Y ya teniendo estos conocimientos empezamos a programar con todo esto e hicimos varios programas como este :

Leer un arreglo de 15 números. Mostrar el arreglo, la suma de los pares y el promedio de los impares.

```
serie con switch real.cpp  serie con switch.cpp  ejercicio 3 de la serie .c  otra forma de serie con switch

#include<stdio.h>
#include<stdio.h>
main()
{
    int op;
    int t,j;
    float b[100];
    int a[15],r,suma=0,c=0;
    float prom,si;
    int i=0,n=0,co=0;
    float q;
    printf("programa de sammy\n");
    printf("selecciona una de las siguientes opciones:\n1) leer 15\n");
    scanf("%d",&op);
    switch(op)
    {
        case 1:
            printf("dame 15 numeros\n");
            for(r=0;r<15;r++)
            {
                scanf("%d",&a[r]);
            }
            printf("su arreglo es:\n");
            for(r=0;r<15;r++)
            {
                printf(" %d\n",a[r]);
            }
            for(r=0;r<15;r++)
            {
                ...
            }
        }
    }
```

```

    {
        if(a[r]%2==0)
        {
            suma=suma+a[r];
        }
        else
        {
            c++;
            si=si+a[r];
        }
    }
    prom=si/c;
    printf("la suma de los pares es: %d\n " , suma);
    printf("el promedio de los impares es: %f" ,prom);
    break;
    case 2:
    printf("¿cuantos numeros quieres que tenga tu arreglo?\n");
    scanf("%d",&j);
    printf("ahora ingresa %d numeros\n",j);
    for(t=0;t<j;t++)
    {
        scanf("%f",&b[t]);
    }
    printf("tu arreglo es: \n");
    for(t=0;t<j;t++)
    {
        printf("%.2f\n",b[t]);
    }
}

```

```

    {
        printf("%.2f\n",b[t]);
    }
    break;
    case 3:
    printf("cuantos numeros quieres que tenga tu arreglo\n");
    scanf("%d",&n);
    float s[n];
    printf("ahora dame un numero\n");
    scanf("%f",&q);
    printf("ahora ingresa %d numeros que escogiste para el tamaño de tu arreglo\n",n);
    for(i=0;i<n;i++)
    {
        scanf("%f",&s[i]);
    }
    for(i=0;i<n;i++)
    {
        if (s[i]==q)
        {
            co++;
        }
    }
    printf (" repitiste el numero %.2f : %d veces",q,co);
}
}

```

```

programa de sammy
selecciona una de las siguientes opciones:
1) leer 15 num y te dire la suma de los pares y el promedio de impares
2) te dire cuantos numeros quieres que lea y te los muestre
3) me diras cuantos numeros quieres que lea, y me diras un numero que te dire cuantas veces lo ingresaste
1
dame 15 numeros
2
1
1
1
2
2
3
3
3
2
2
2
3
1
1
1
su arreglo es:
2
1
1
1
1
2
2
2
3
3
3
2
3
1
1
1
1
la suma de los pares es: 10
el promedio de los impares es: 1.800000

```

Esto me ayudo a practicar mucho el tema de arreglos que me llevo a costar un poco, y después del tema de arreglos vimos funciones que son funciones que uno mismo como programador crea, vimos de 2 tipos con retorno y sin retorno pero sinceramente mi maestra no supo enseñar bien este tema y tuve que estudiar por mi cuenta, después de eso paso lo de la pandemia y la maestra solo nos mando archivos del tema de archivos y apuntadores y desgraciadamente no lo enseñó ni lo pudimos poner en practica y esos temas se me complican mucho.

Finalmente creo que tuve un curso regular de fundamentos de programación donde los últimos temas mas importantes por cuestiones de la pandemia no logre obtener los conocimientos necesarios de esos últimos temas.

