

Tarea 8

Luna Olivares Mario Andrés
Numero de Cuenta 411012353
Profesora: Elizabeth Fonseca
Grupo: 1129
Fecha: 2010-10-02

Resumen de clase.

Características de los algoritmos:

1. • Precisión
2. • Simplicidad
3. • Finitud
4. • Carácter mecánico
5. • Procedimiento general

No podemos realizar un programa si antes hacer un pre-programa a mano en el cual debemos ver las funciones básicas del mismo. Y debemos contemplar que los compiladores no empiezan hasta que un proceso termina.

En los programas hechos en C se sigue la siguiente estructura.

Cabeceras: sirven para indicar el tipo de comandos que se pueden utilizar dentro del programa, van seguidas del signo # y un include de la siguiente manera.

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

Programa principal. Es donde están las órdenes que debe realizar, estas ordenes se escriben en forma de comandos, que como se mencionó anteriormente dependen del tipo de cabecera que se este utilizando. Se coloca la palabra main paréntesis y abajo dos llaves una de apertura y otra de cierre, dentro de estas llaves se incluyen los comandos. Por ejemplo:

MAIN:()

{

Orden Ejemplo 1;
Orden Ejemplo 2;
Orden Ejemplo 3;

Al final de cada orden se agrega; para que realice la acción ingresada.

Orden "N"

}

Programa Secundario.-Este incluirá una o varias funciones, pero lo importantes es que solo tiene un programa principal.

Programa

Función 1 ()

{ }

Función 2 ()

{ }

Tipo de programación.

Programación no estructurada: Son programas con comandos conocidos.

Programación estructurada: Se maneja en secuencia, selección y repetición.

En general un programa tiene la siguiente estructura.

```
main()
{
printf("Ya pase");
getch();
system("cls");
getch();
}
```

El comando printf (imprimir archivo); sirve para mostrar un letrero en la pantalla y getch(); para esperar a que se oprima una tecla para borrar la pantalla o que aparezca un nuevo letrero.

El comando system (cls); sirve para limpiar pantalla, en otros compiladores como los son borlandc su equivalente es cls scr();

La secuencia de programas

Un programa puede hacerse de dos formas diferentes con diagrama de flujo y en lenguaje c:

Ejemplos:

Diagrama de flujo

En su estructura tiene figuras en las que se encuentra la orden y con flechas se indica el siguiente comando.



Lenguaje C
Su secuencia es como la mencionada anteriormente.

```
main()
{
printf("Ya pase");
getch();
system("cls");
getch();
}
```

Selección o condicional

El programa selecciona lo que debe ejecutar, de acuerdo con las ordene. En lenguaje c y diagrama de flujo utiliza if simple, if doble y switch.

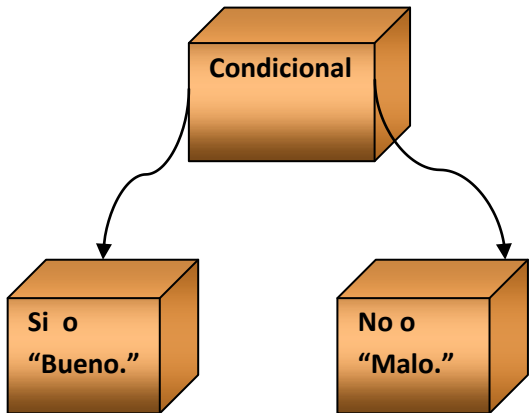
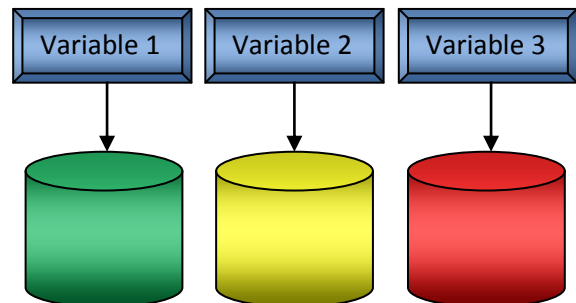
<p>Diagrama de flujo.</p>  <pre> graph TD A[Condional] --> B["Si o \"Bueno.\""] A --> C["No o \"Malo.\""] </pre>	<p>Lenguaje C</p> <p>En C también hay if simple, doble y switch de la siguiente manera</p> <pre> if (x>y) Printf("x es mayor"); Printf("y es mayor"); </pre> <p>El if simple seria cuando esto es verdadero, el if doble cuando es falso y es switch cuando hay varios casos.</p>
---	--

Diagrama de flujo.

Algunas de las condiciones que puede incluir son:

- a=b
- b<a
- b>a
- b≠a
- a or b
- a and b

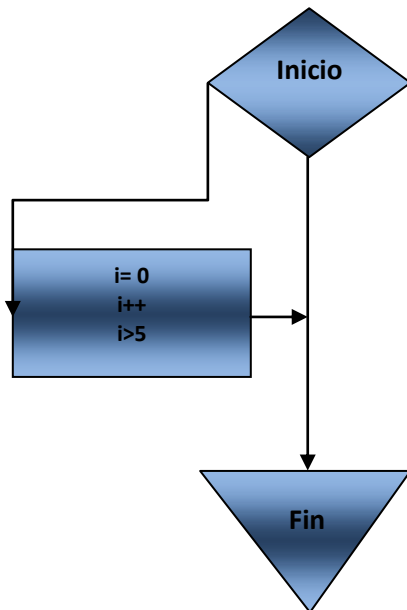
En el caso de Switch hace una combinación de variables, haciendo uno u otro.



Repetición

Es utilizado para operaciones que van a ser infinitas a menos que se especifique lo contrario. Puede utilizar las funciones for, while o do-while.

Diagrama de flujo.



`i=0` indica el inicio
`i++` `i=i+1` se conoce como contador
`i>5` condicional.

Lenguaje C

```
#include <stdio.h>
int a
main ()
{
printf(A poco no me quedo de
10);
}
```