

PRACTICA No 1

Objetivos generales: Que el alumno aplique los conocimientos básicos obtenidos en la materia de programación para sentar las bases del desarrollo de programas mas complejos.

Desarrollo:

La práctica que en esta ocasión realizamos fue la creación de 5 distintos programas dos de ellos aprendidos en la clase de teoría, uno de ellos solo para transcribir al programa, uno al que había que corregirle los errores y uno mas que dadas las condiciones habria que desarrollar.

El primer programa fue estudiado en clase y su objetivo era el de calcular el área de un circulo mediante la inserción del radio del mismo por el usuario.

El programa que estudiamos en clase tenia algunas omisiones de comandos los cuales eran:

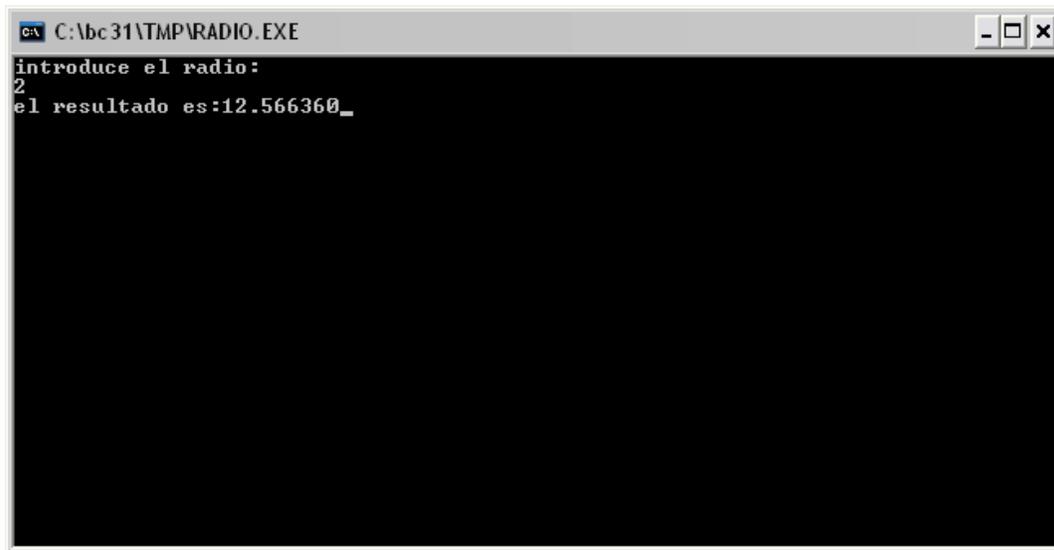
La librería conio que sirve para hacer cálculos matemáticos simples.

El comando clrscr que es el comando que nos sirve para limpiar la pantalla.

El comando getch que es el encargado de sacarnos del programa realizado mediante un enter.

Al final el programa quedo de la siguiente forma:

```
/*
PROGRAMA: 1
OBJETIVO: APLICAR LOS CONOCIMIENTOS
DE LENGUAJE C PARA APLICARLOS A LA
REALIZACION DE UN PROGRAMA QUE OBTENGA
EL AREA DE UN CIRCULO*/
#include <stdio.h>
#include <conio.h>
main ()
{
float r, area;
const float pi=3.14159;
clrscr ();
printf ("introduce el radio:\n");
scanf ("%f", & r);
area= pi*r*r;
printf ("el resultado es:%f", area);
getch ();
}
/*CONCLUSIONES:EN ESTE PROGRAMA APLICAMOS
LA SECUENCIA QUE SE DEBE LLEVAR A CABO PARA
OBTENER UN PROGRAMA FUNCIONAL*/
La corrida de este programa fue el siguiente:
```



```
C:\bc31\TMP\RADIO.EXE
introduce el radio:
2
el resultado es:12.566360_
```

Realizaba de forma correcta su función y hay que agregar que parte de esta práctica fue el aprender algunas cosas que se deben de hacer cuando se esta realizando un programa en lenguaje C.

Estas cosas que se deben hacer y que realizamos son:

Primero que nada si ya llevamos un avance en nuestro programa debemos salvarlo y cada vez que avancemos un poco volverlo a hacer.

Cuando se cree que se ah terminado el programa se debe de compilar para que el compilador de C revise la sintaxis de nuestro programa y podamos estar seguros de que por esta parte no tendremos problemas. Si hay errores debemos de ver en que línea se encuentran y revisar arriba y debajo de esta para verificar el error y corregirlo.

Ya que hemos hecho esto podemos ejecutar o correr el programa con el comando run en la barra de herramientas.

Después de hacer esto nos aparecerá una pantalla negra en la cual mediante el comando printf visualizaremos lo que quisimos poner en nuestro programa. Si insertamos algún dato mediante el comando scanf y la dirección y tipo de dato lo leerá nuestro programa y con esto bastara para que haga una operación y nos muestre de nuevo mediante el comando printf y la variable el resultado para el que esta programado con otro enter regresamos al programador de C.

En el segundo programa se trataba de que mediante la inserción de tres números se visualizara la suma de estos mismos por el usuario.

El programa quedo de la siguiente forma:

```
/*
PROGRAMA 2
OBJETIVO: REALIZAR DE MANERA CORRECTA
UN PROGRAMA QUE SUME TRES VARIABLE*/
#include <stdio.h>
#include <conio.h>
main ()
```

```

{
float suma, x, y, z;
clrscr ();
printf ("introduce el valor 1:\n");
scanf ("%f", & x);
printf ("introduce el valor 2:\n");
scanf ("%f", & y);
printf ("introduce el valor 3:\n");
scanf ("%f",& z);
suma=x+y+z;
printf ("El resultado es:%f", suma);
getch ();
}
/*CONCLUSIONES: EN ESTE PROGRAMA REAFIRMAMOS QUE MUCHOS DE
LOS CAMANDOS DEL
LENGUAJE C SON REPETITIVOS Y QUE HAY ALGUNOS INDISPENSABLES
Y OTROS COMO GETCH QUE NOS AYUDAN A DARLE PRESENTACION*/

```

De nuevo el programa funciono de manera correcta ya que realizamos todos los pasos necesarios y nótese que ocupamos los mismos comandos y librerías del programa anterior excepto la suma que en este caso fue de tres variables que fueron debidamente declaradas junto con la variable suma.

La corrida de este programa es la siguiente:

```

C:\abc 31\TMP\SUMA.EXE
introduce el valor 1:
2
introduce el valor 2:
2.1
introduce el valor 3:
3.4
El resultado es:7.500000

```

El tercer programa solo fue la transcripción de uno que la maestra nos proporciono el propósito de este programa era el de conocer la edad de una persona conociendo el año en que nació.

El programa quedo de la siguiente manera:

```

/*
PROGRAMA 3
OBJETIVO: EN ESTE PROGRAMA SE APRENDERA EL USO DE
LAS VARIABLES GLOBALES ASI COMO DEL USO DE NUMEROS ENTEROS*/

```

```

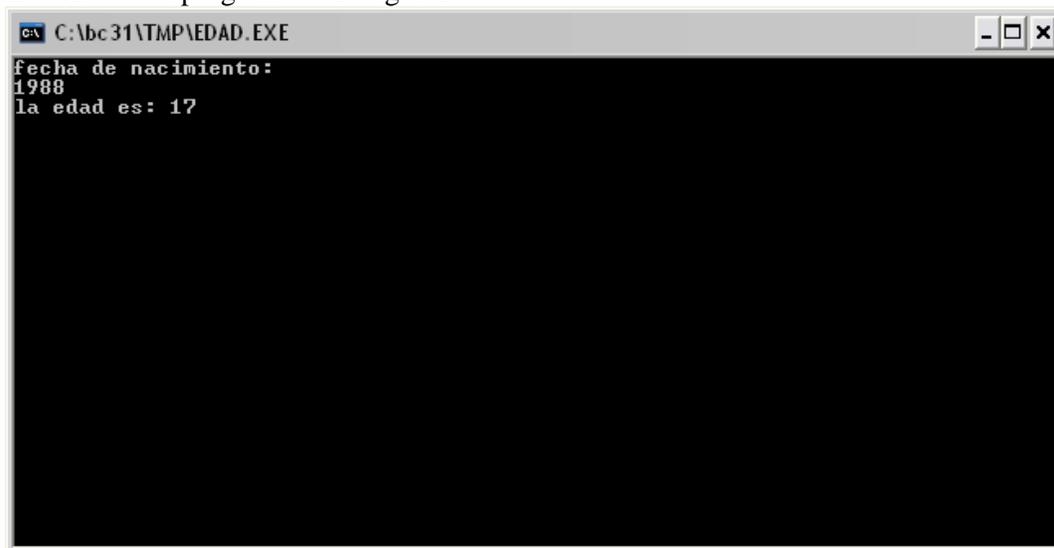
#include <stdio.h>
#include <conio.h>
#define actual 2005
main ()
{
int nac, edad;
clrscr ();
printf ("fecha de nacimiento:\n");
scanf ("%d", &nac);
edad=actual-nac;
printf ("la edad es: %d", edad);
getch ();
}
/*CONCLUSIONES:EN ESTE PROGRAMA APRENDIMOS LA
FORMA DE USAR OTRAS VARIABLES Y EN GENERAL LA ESTRUCTURA
DE LOS COMANDOS PRINTF Y SCANF*/

```

Este programa también funciono de manera correcta pero agregaría que para evitar confusiones en vez de informar al usuario de que escriba su fecha de nacimiento creo que debería decir que insertara su año de nacimiento además como nota este programa sacaba la edad con respecto a 2005.

También hay que destacar dos cosas, la primera es que usamos una variable global que esta escrita como #define 2005, y la segunda que en este ejercicio me di cuenta de que cuando salvas el programa en la maquina aparecen dos cosas primero un ejecutable de el programa también el programa en Word pad.

La corrida del programa es la siguiente:



El cuarto programa fue la corrección del código fuente de un programa que se nos proporciono el cual es el siguiente:

```

#include <stdio.h>
#include <conio.h>
#define intereses 0.18

```

```

main ()
{
float lcapital, inv_tot;
clrscr ();
printf ("la inversion es:%d");
scanf ("%lf", &lcapital);
inv_tot =lcapital-(lcapital*intereses);
printf ("la inversion es : %d", inv);
getch ();
}

```

Este programa lo realice lo guarde y lo compile. El resultado de la compilacion fueron los siguientes mensajes de error:

```

Error INTERESES.C 6: Declaration terminated incorrectly
Error INTERESES.C 9: Must take address of a memory location
Error INTERESES.C 9: Function call missing )
Error INTERESES.C 10: Underfined symbol 'inv_tot'

```

Para solucionar el primer problema lo que hice fue quitar el 1 de la variable lcapital pues dentro de las reglas esto no es permitido y generaba un error .
Después de hacer los cambios compile de nuevo y ahora los nuevos errores fueron:

```

Error INTERESES.C 11: Underfined symbol 'inv'
Warning INTERESES.C 13: 'inv_tot' is assigned avalue that is never used

```

Para solucionar los problemas complete la variable que estaba como inv por inv_tot. Luego procedí a compilar de nuevo y no se encontraron errores guarde y ejecute el programa en esta parte lo que sucedió fue que en la pantalla apareció: "la inversion es: 7914 y al poner algún numero y dar un enter lo que aparecía es: "la inversion es: 0" al dar otro enter aparecía una ventana de dialogo que decia La cpu NTVDM ha encontrado una instrucción no permitida es: 89ª IP:b2292 OP: ff ff eb05 elija cerra para finalizar la aplicación.

Por lo que el programa estaba mal, para corregir este error lo que realice fue cambiar el neumónico de las variables ya que en printf se declaraba una de valor entero que no existía y en scanf se quería obtener uno de valor double por lo que las cambie ambas a las de valor de coma flotante.

Otra corrección esta de tipo interpretativo seria que en vez de pedir que el usuario introduzca la inversión debe pedir que introduzca el capital. El programa tendría entonces una aplicación especifica de obtener la inversión que se hace sobre un bien por el cual se pagan inicialmente un 18 % de interés.

Con lo cual el programa que quedo fue el siguiente:

```

/*:
PROGRAMA 5
OBJETIVO: APRENDER A INTERPRATAR EL TIPO DE
ERRORES QUE SE PUEDEN COMETER EN EL DISEÑO DE UN PROGRAMA*/
#include <stdio.h>

```

```

#include <conio.h>
#define intereses 0.18
main ()
{
float capital, inv_tot;
clrscr ();
printf ("El capital es:\n");
scanf ("%f", &capital);
inv_tot =capital-(capital*intereses);
printf ("la inversion es : %f", inv_tot);
getch ();
}
/*CONCLUSIONES: EN ESTE PROGRAMA APRENDIMOS A RECONOCER LOS
ERRORES QUE
SE COMETEN GENERALEMENTE EN UN PROGRAMA TANTO DE SINTAXIS
COMO DE
CORRIMIENTO ADEMAS DE ASEGURAR EL FUNCIONAMIENTO DE
COMPILAR QUE ES
MUY IMPORTANTE*/

```

Este programa también hacia correctamente su función. Y su corrida es la que sigue:

```

C:\abc 31\TMP\CORRECCI.EXE
El capital es:
100000
la inversion es : 82000.000000_

```

La última parte fue la de realizar un programa partiendo del siguiente enunciado:

Encontrar la pendiente de la siguiente ecuación $y=mx + b$

- 1.- Identificamos variables de entrada y de salida
- 2.- Especificando el método a usar.

Lo que hice fue primero ver como encontrar la pendiente y me encontré con la formula de la pendiente es igual al cociente de la diferencia de las absisas de dos puntos entre las ordenadas de estos mismos puntos.

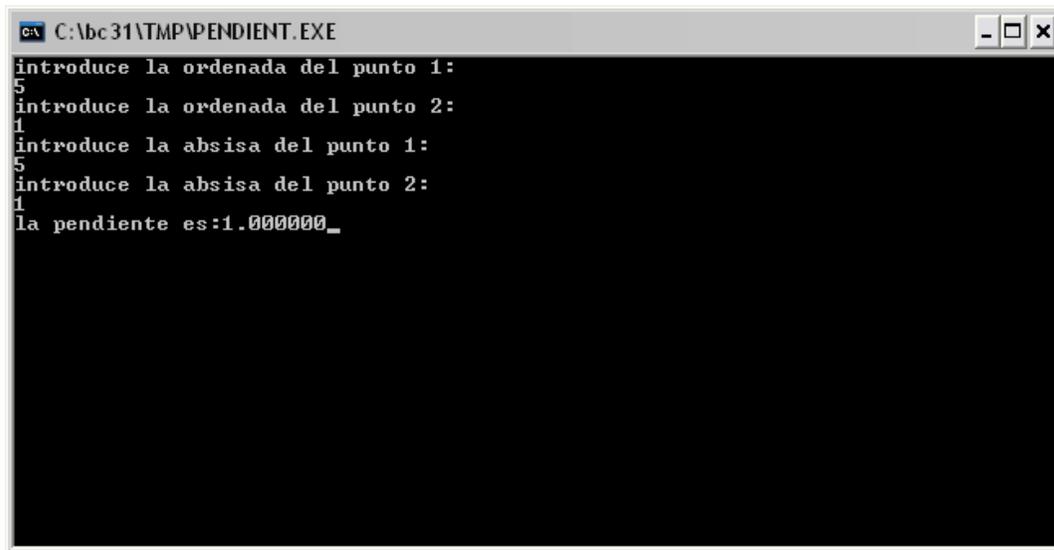
Por lo que tendria 5 variables x_1 , x_2 , y_1 , y_2 y pendiente que esta ultima seria el resultado.

Por lo tanto el programa quedo de la siguiente forma:

```
/*
PROGRAMA 5
OBJETIVO:REALIZAR UN PROGRAMA RECONOCIENDO LAS VARIABLES Y
LA ESTRUCTURA LOGICA QUE DEBE USARSE PARA PODER OBTENER EL
RESULTADO DESEADO*/
#include <stdio.h>
#include <conio.h>
main ()
{
float m,x1,x2,y1,y2;
clrscr ();
printf ("introduce la ordenada del punto 1:\n");
scanf ("%f", &x1);
printf ("introduce la ordenada del punto 2:\n");
scanf ("%f", &x2);
printf ("introduce la absisa del punto 1:\n");
scanf ("%f", &y1);
printf("introduce la absisa del punto 2:\n");
scanf ("%f",&y2);
m=(y2-y1)/(x2-x1);
printf ("la pendiente es:%f", m);
getch ();
}
/*CONCLUSIONES: EN ESTA PRACTICA APRENDIMOS A REALIZAR
UN PROGRAMA DADOS SOLO LOS REQUERIMIENTOS DE ESTE
ASEGURANDO
ASI UN APRENDIZAJE SIGNIFICATIVO QUE ADEMAS NOS MUESTRA QUE
PODEMOS REALIZAR PROGRAMAS ADECUADOS A NUESTRAS
NECESIDADES*/
```

El programa encuentra la pendiente de una recta dados dos puntos que estén en esta recta. Hay que decir que los valores de los puntos deben ser enteros.

La corrida de este programa es la que sigue:



```
C:\abc31\TMP\PENDIENT.EXE
introduce la ordenada del punto 1:
5
introduce la ordenada del punto 2:
1
introduce la absisa del punto 1:
5
introduce la absisa del punto 2:
1
la pendiente es:1.000000_
```

Conclusiones generales:

En esta practica se cumplió el objetivo de usar lo aprendido en teoría sobre la programación en lenguaje c puesto que aprendí a usar las variables y a realizar un programa de buena manera ya que todos los programas sirvieron para su propósito hay cosas muy importantes como no olvidar que primero se deben poner las librerías que vamos a usar luego la función principal main() la apertura y el fin del programa que se simboliza con los corchetes, declarar las variables que vamos usar y luego comandos como printf y scanf que nos dan comunicación con el usuario y las funciones aritméticas que deben de llevar que tipo de variable manejan además de comandos importantes como getch y clrscr. También aprendí a compilar y salvar el programa. En general fue una muy buena forma de aprender de forma práctica lo que no sabia acerca de los programación en lenguaje C.