|  |  |  |
| --- | --- | --- |
|  | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Rodriguez Espino Claudia. |
| *Asignatura:* | Fundamentos de Programacion |
| *Grupo:* | 1102 |
| *No de Práctica(s):* | Práctica 8:Estructuras de Selección |
| *Integrante(s):* | Martínez Martínez Yanni |
|  |  |
|  |  |
| *Semestre:* | 2018-1 |
| *Fecha de entrega:* | 14 de Octubre de 2017 |
| *Obervaciones:* |  |
|  |  |

**Práctica 8: Estructuras de selección**

**Objetivo:**

Elaborar programas en lenguaje C que incluyan las estructuras de selección if, if-else, switch y ternaria (o condicional) para la resolución de problemas básicos.

**Actividades:**

* Elaborar expresiones lógicas/condicionales utilizadas en las estructuras de selección y realizar su evaluación.
* Elaborar un programa en lenguaje C para cada estructura de selección.

**Desarrollo:**

En la presente práctica se vieron algunas de las funciones de comandos que ya se conocian, como lo es el if, if-else y switch case. Pero también se revisaron otros nuevos comandos los cuales no se habian visto con anterioridad, algunos son:

* **Enumeración:** Se utiliza el comando “enum” y debe ser declarada como constante. Esto sirve para crear numeraciones dentro del programa.
* **Estructura de control selectiva condicional:** Este comando realiza una comparación rápida y si se cumple la condición ejecuta una instrucción, y en caso contrario ejecuta otra.

Dicho esto, de forma grupal se hizo énfasis a conocer cual es el uso de estos realizando algunos programas los cuales serán anexados a la presente práctica como se verán a continuación, haciendo uso adecuado de los nuevos conceptos planteados, conocer su correcto uso y ampliar la forma de programar al momento, brindando nuevas alternativas a utilizar con programas futuros.

Formula con estructura de control selectiva:

#include <stdio.h>

int a,b,c,d,y,x;

main()

{

printf("ingresa el valor de 'y' disinto de 2 \n");

scanf("%d" ,&y);

if(y==2)

do

{

printf("introduce otro valor \n");

scanf("%d" ,&y);

}

while(y==2);

a=y\*y;

b=4\*y;

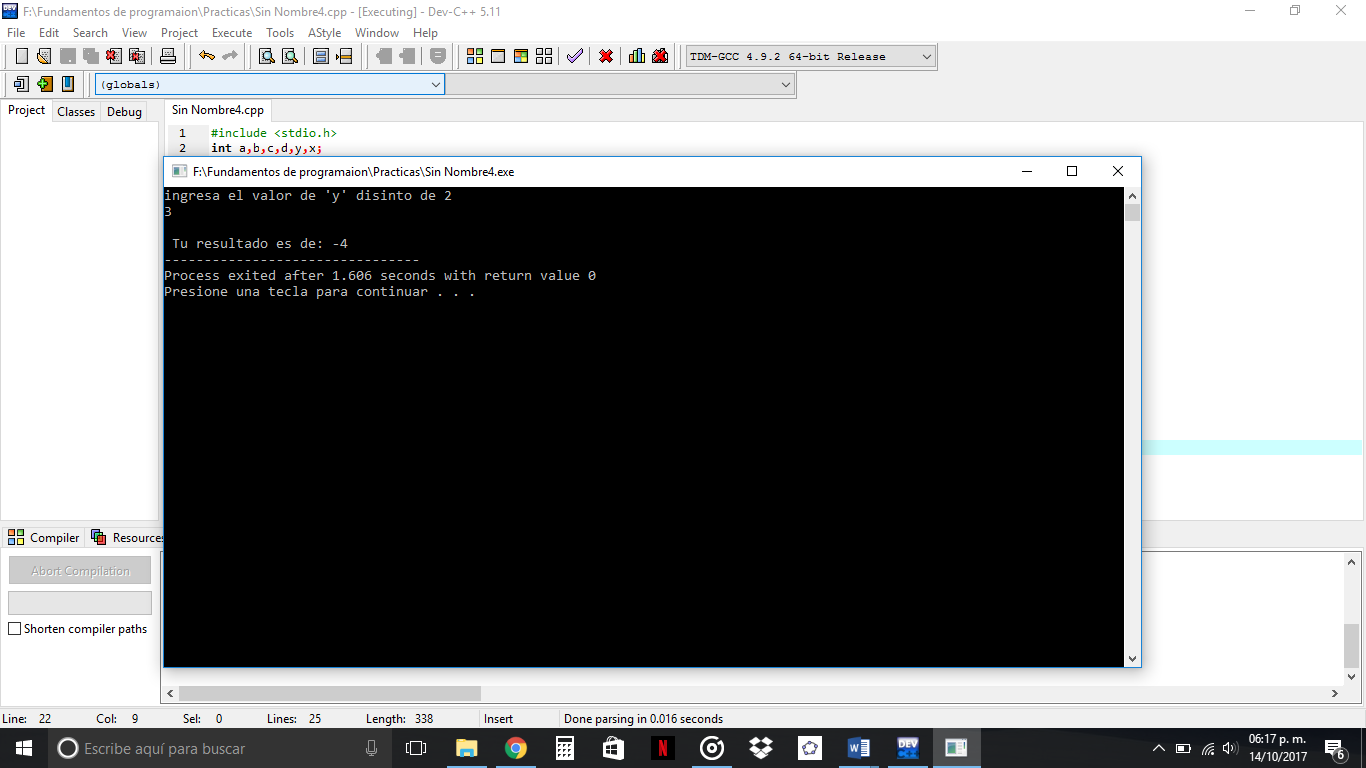
c=4\*a;

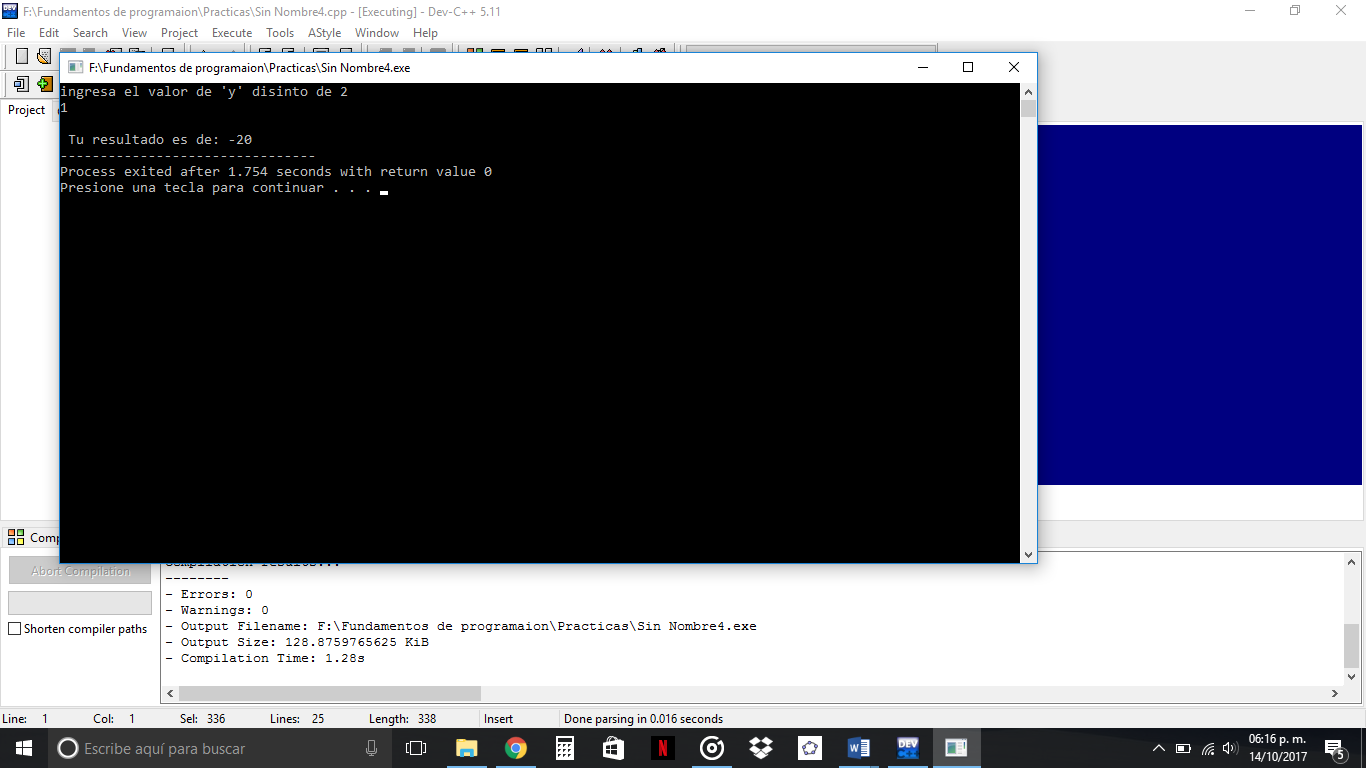
d=3\*y;

x=y>2?a+b-25:c-d;

printf("\n Tu resultado es de: %i" ,x);

}





Menú con números:

#include<stdio.h>

int op;

main()

{

printf("\t Menu \n\n");

printf("Elegir la opcion deseada\n\n");

printf("1) Altas \n");

printf("2) Bajas \n");

printf("3) Cambios \n");

scanf("%c",&op);

switch(op)

{

case '1':

printf("Se selecciono 'Altas'.\n");

break;

case '2':

printf("Se selecciono 'Bajas'.\n");

break;

case '3':

printf("Se selecciono 'Cambios'.\n");

break;

default:

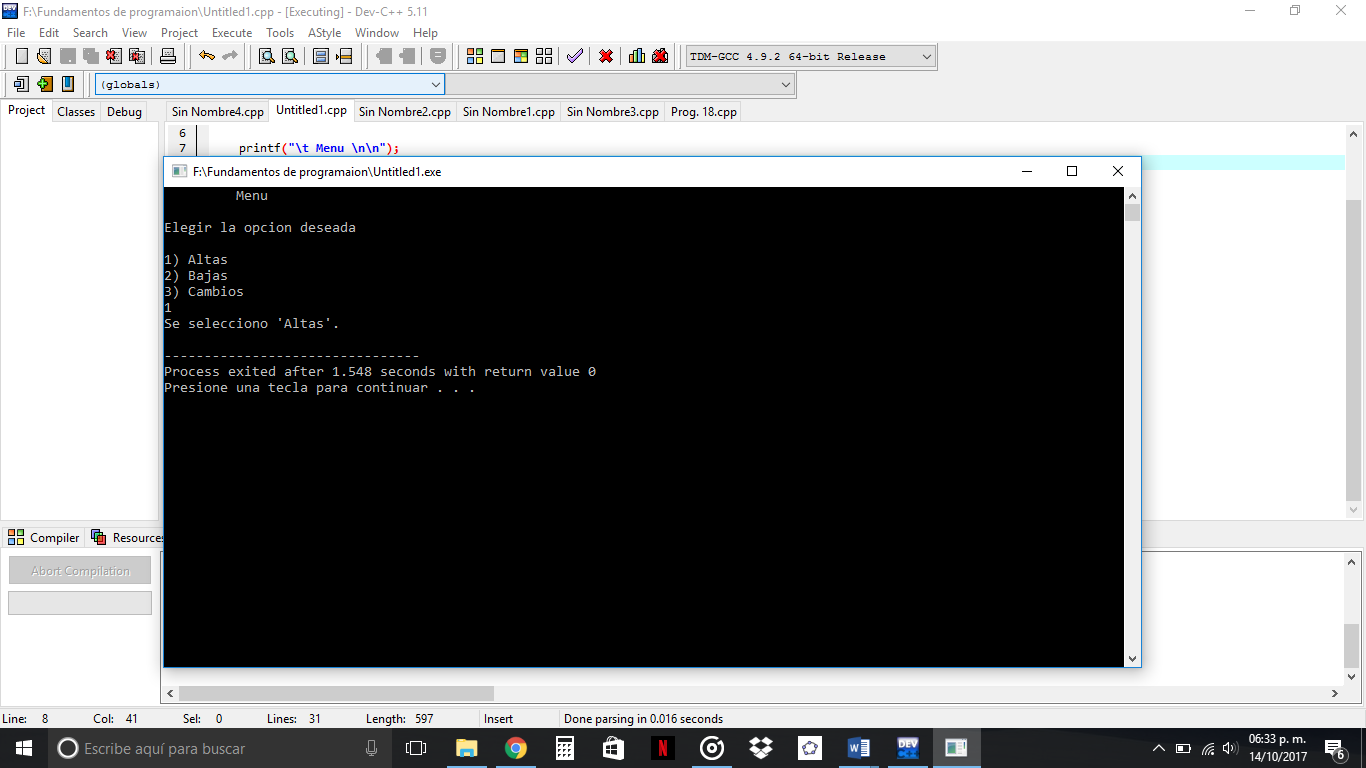
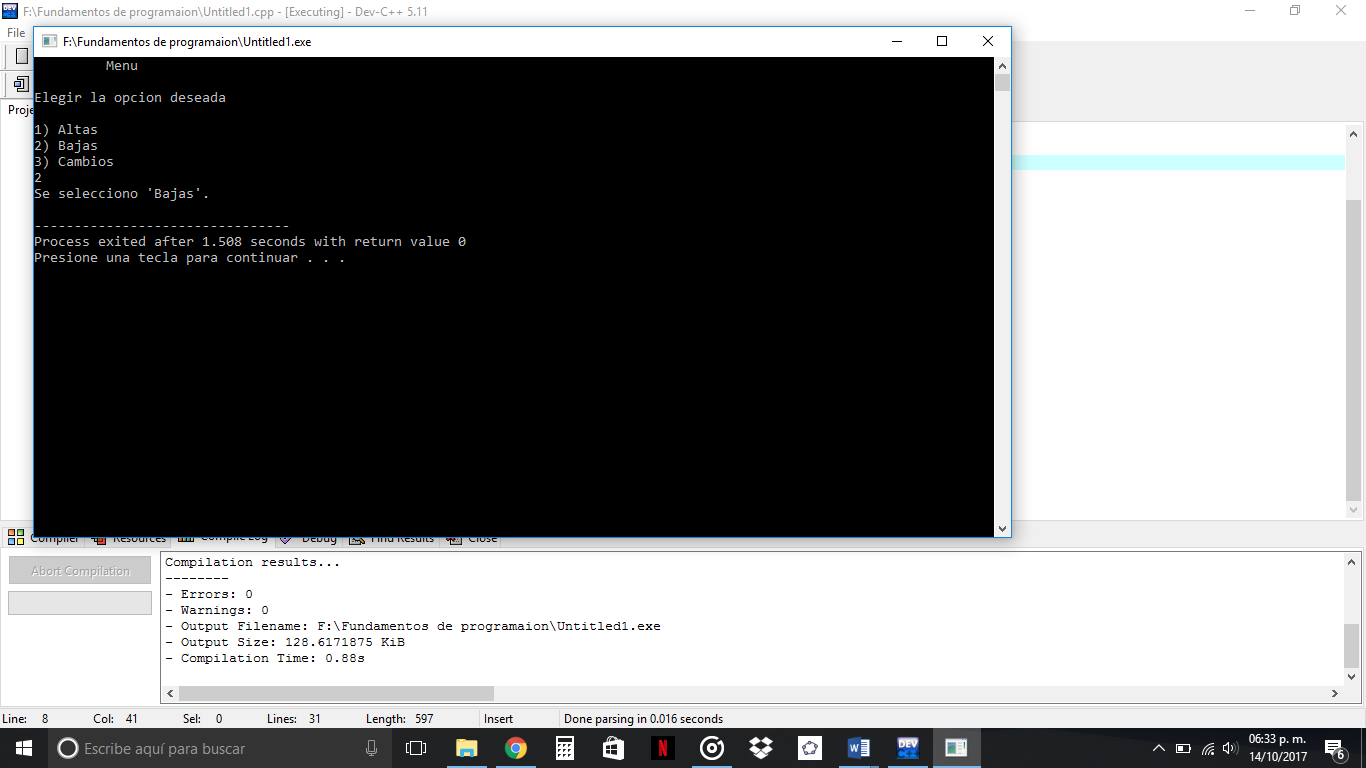
printf("Opcion no valida.\n");

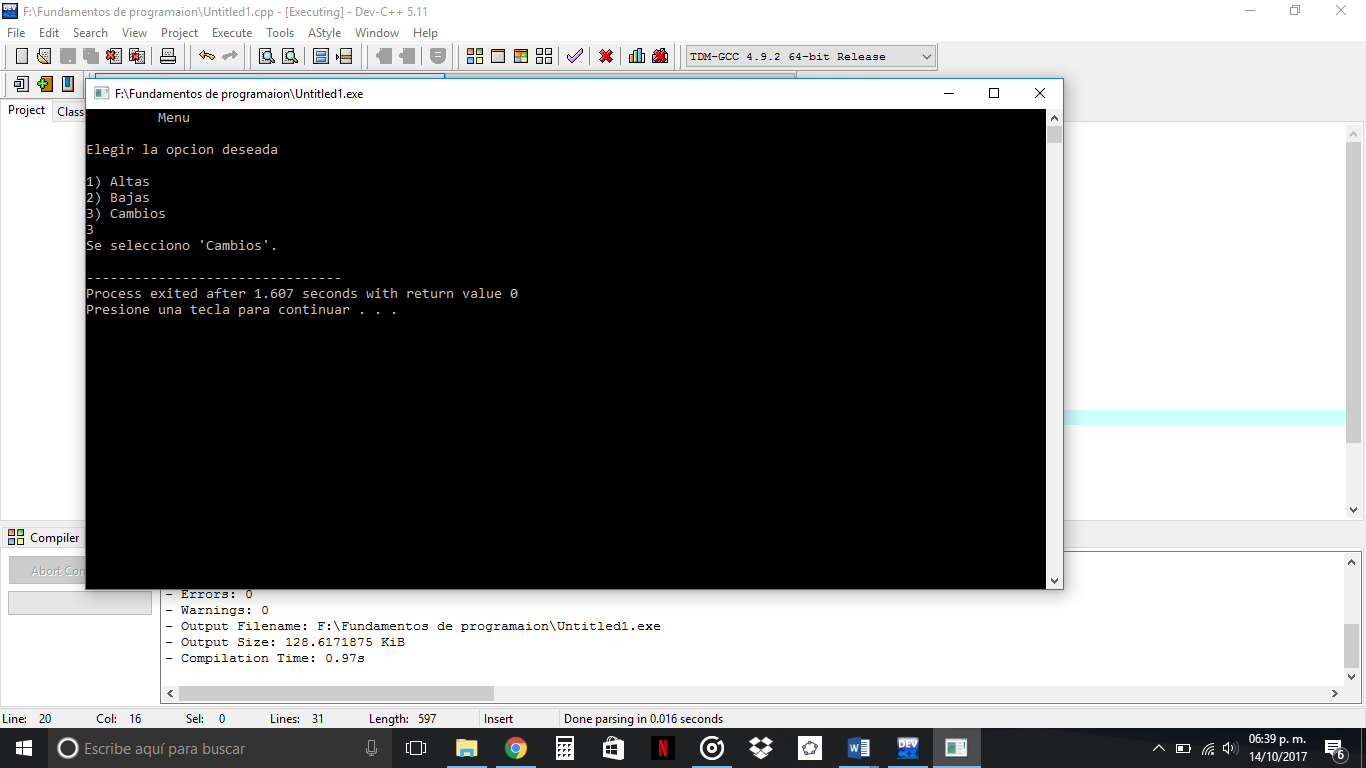
break;

}

return 0;

}



Menú con caracteres:

#include<stdio.h>

#include<ctype.h>

char op='\0';

main()

{

printf("\t Menu \n\n");

printf("Elegir la opcion deseada\n\n");

printf("a) Altas \n");

printf("b) Bajas \n");

printf("c) Cambios \n");

scanf("%c",&op);

op=tolower(op);

switch(op)

{

case 'a':

printf("Se selecciono 'Altas'.\n");

break;

case 'b':

printf("Se selecciono 'Bajas'.\n");

break;

case 'c':

printf("Se selecciono 'Cambios'.\n");

break;

default:

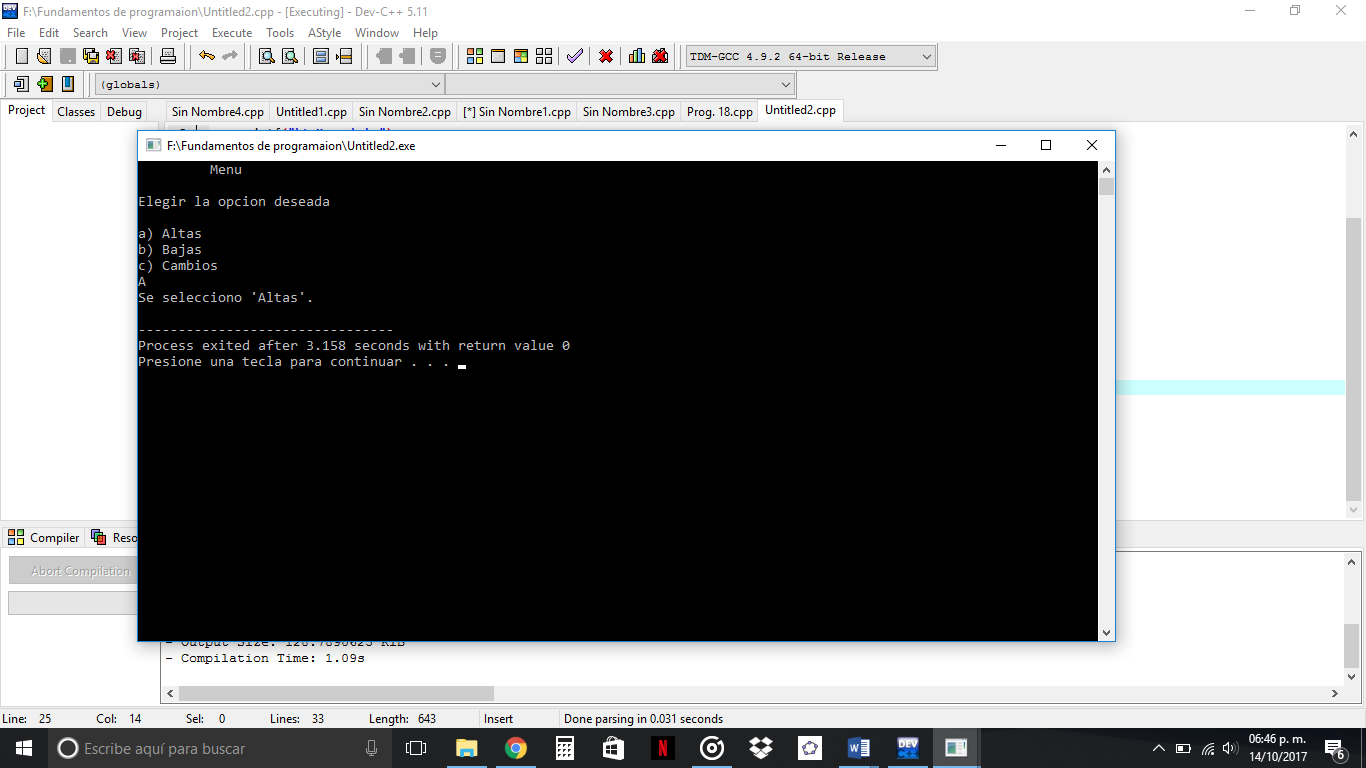
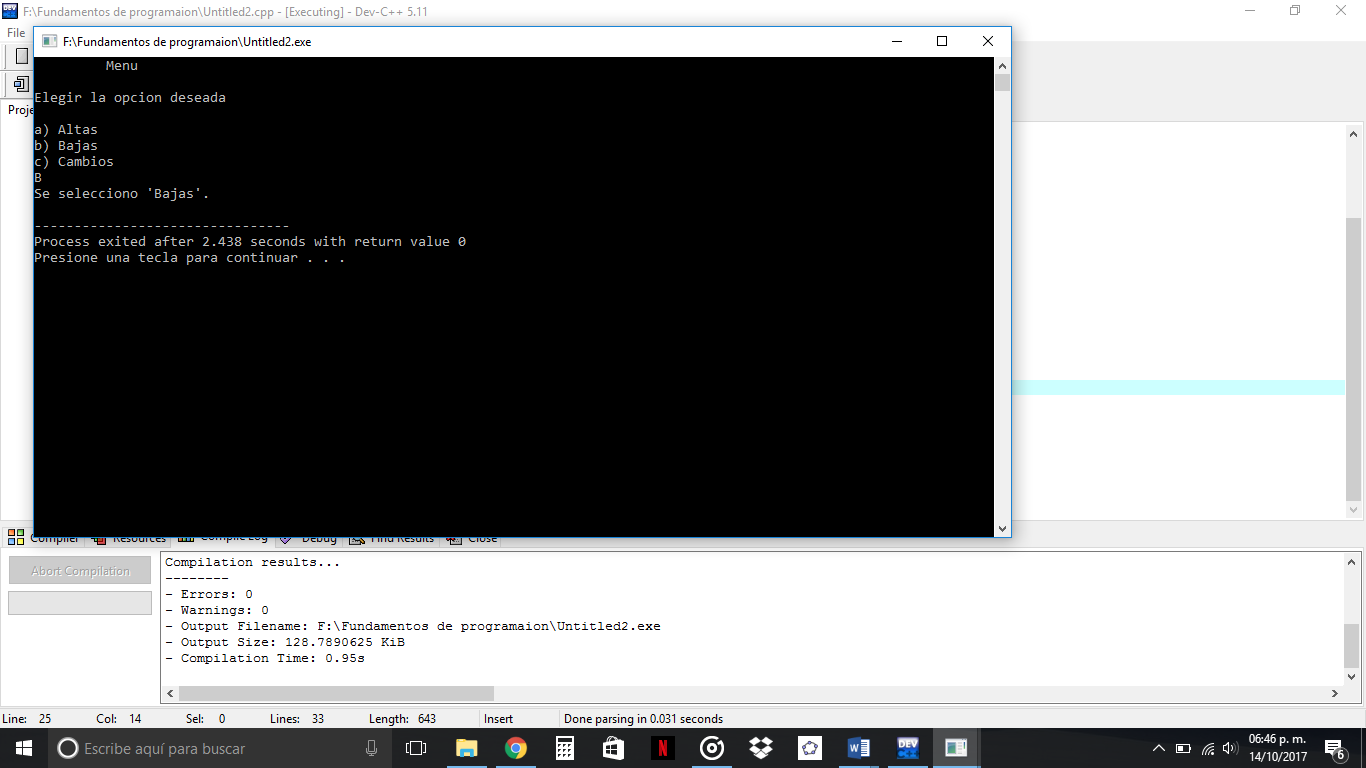
printf("Opcion no valida.\n");

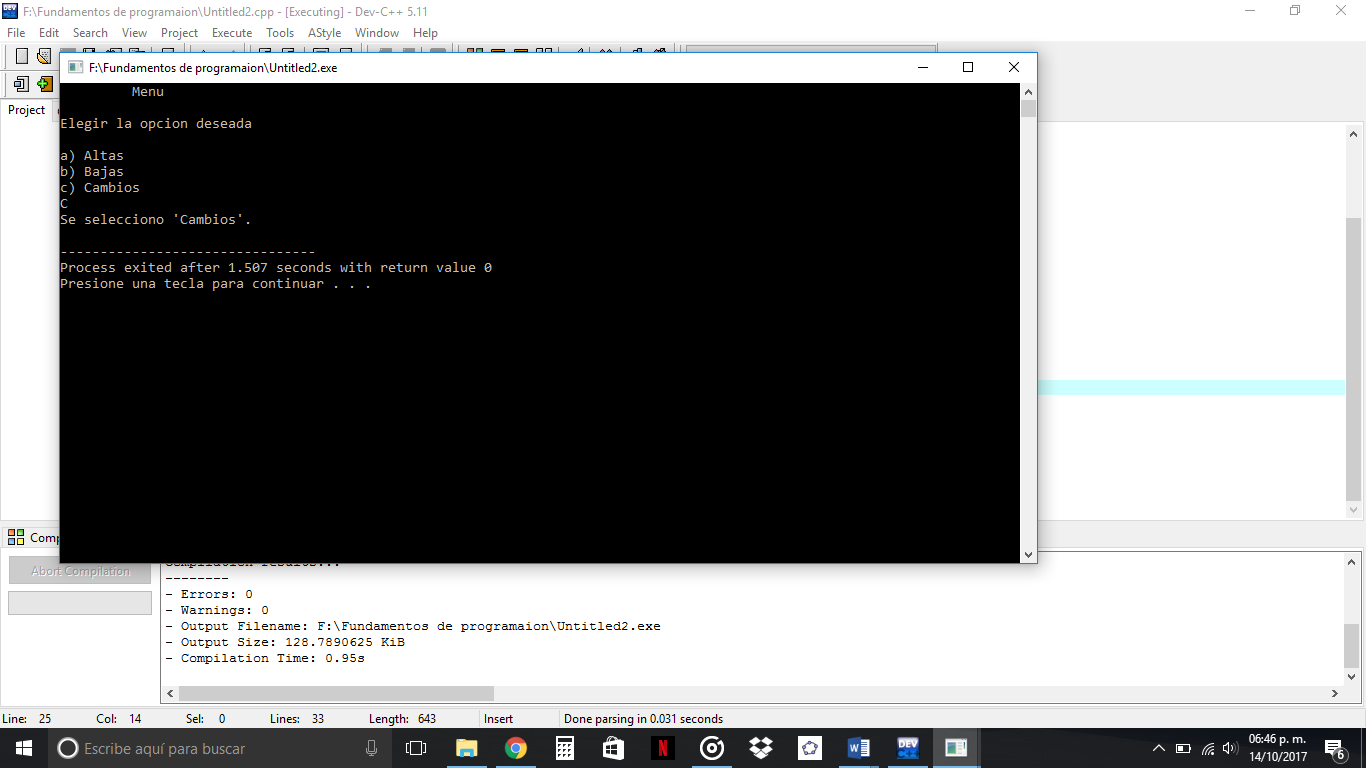
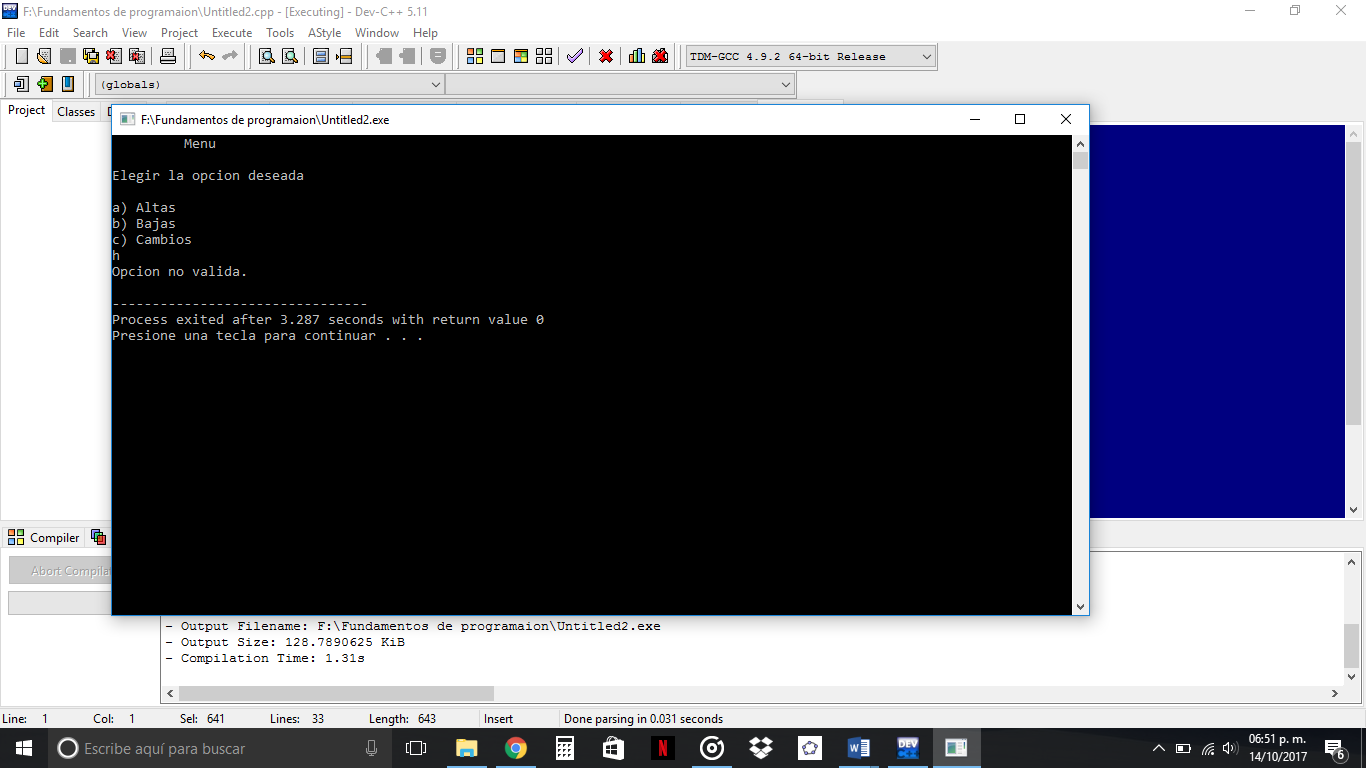
break;

}

return 0;

}

Formula con if-else:

#include<stdio.h>

#include<math.h>

int a=2,s,t,u,v,w,x,y,z;

main()

{

printf("ingresa el valor de ‘y’ distinto a 2 \n");

scanf("%i" ,&y);

if(y<2)

{

x=pow(y,a);

z=4\*x;

w=2\*y;

v=z-w;

printf(“Tu resultado es de: %i" ,v);

}

else

if(y>2)

{

u=pow(y,a);

t=4\*y;

s=u-t+20;

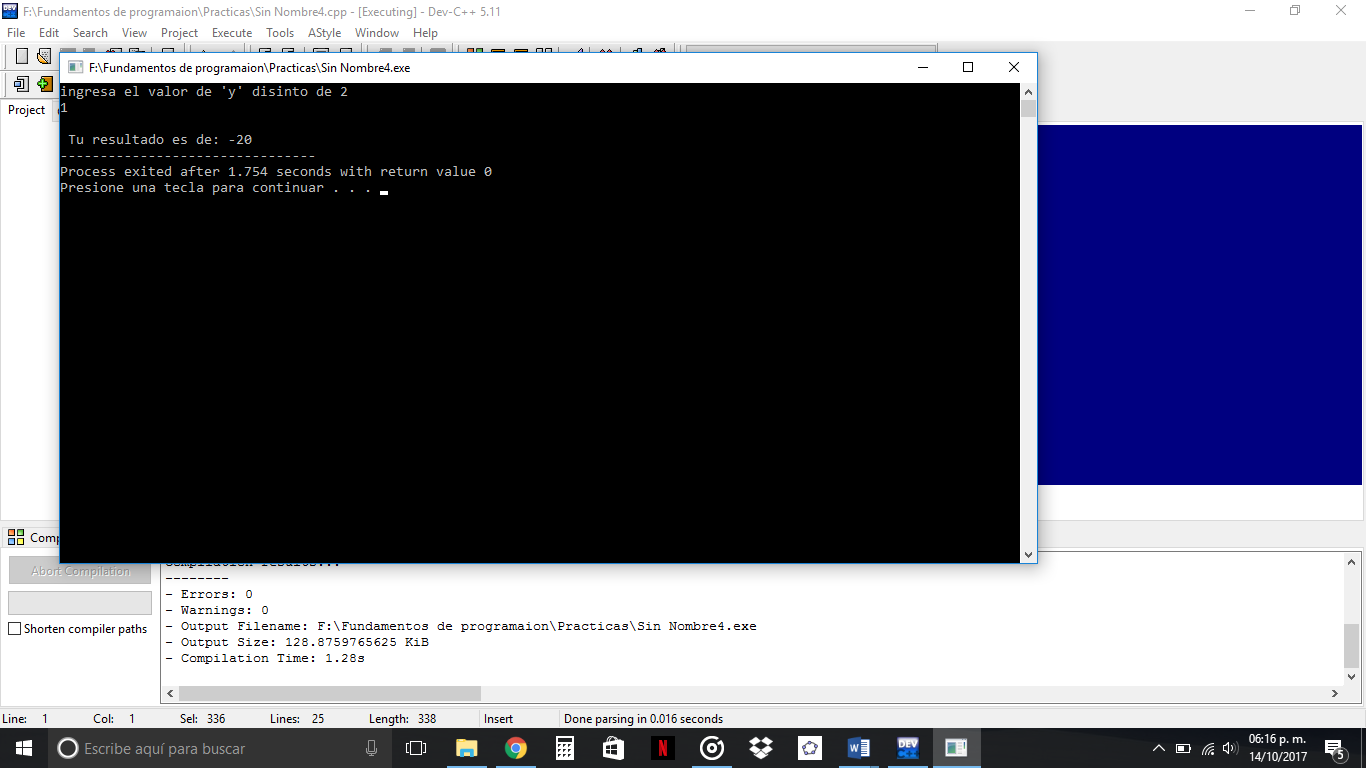
printf("Tu resultado es de: %i" ,s);

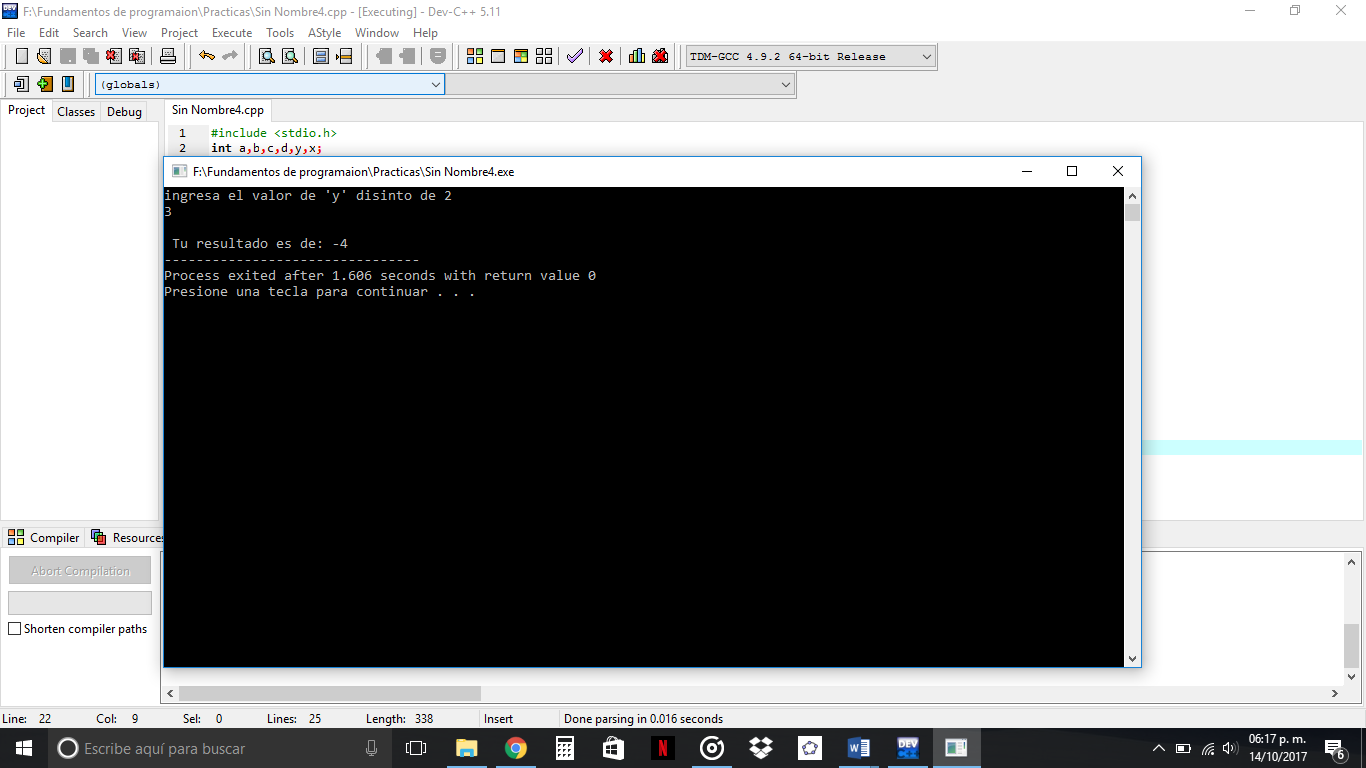
}

else

printf("\n introduce un numero distinto a 2");

}





**Conclusiones:**

En la presente práctica aprendi que la estructura de control selectiva puede servir para sustituir al if-else brindando una major estetica y siendo mas práctico y útil a la vista del usuario aun que se debe saber cuando usar y cuando no, ya que es evidente que no es necesario para programas con pocas opciones o pocas funciones, ademas esta práctica me ayudo bastante a poder perfeccionar el uso de algunos conceptos.