

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

UNIVERSIDAD AUTONOMA METROPOLITANA

Introducción a la Programación

Práctica 2:

“Lenguaje de Programación C: Estructura de Selección Condicional Múltiple y Estructuras de Control Iterativas.”

Fecha de Elaboración:

Martes, 29 de abril de 2008

Fecha de Entrega:

Martes, 13 de mayo de 2008

Contenido

1	Objetivos.....	2
2	Introducción.....	2
3	Desarrollo	3
3.1	Actividad 1 “Estructura de control condicional múltiple: If anidados Vs. Switch”	4
3.2	Actividad 2 “Estructura de Control Iterativa – While (Mientras)”	8
3.3	Actividad 3 “Estructura de Control Iterativa – While (Mientras) – Factorial de un Número”	9
3.4	Actividad 4 “Estructura de Control Iterativa – Do ... While (Hacer ... Mientras) –”	10
4	Resultados.....	11
4.1	Resultados actividad 1	11
4.2	Resultados actividad 2	11
4.3	Resultados actividad 3	11
4.4	Resultados actividad 4	12
5	Cuestionario.....	13
6	Conclusiones.....	14
7	Referencias.	14
8	Actividades Extras.....	14

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

AVISOS:

- La práctica puede desarrollarse con a lo más un compañero.
- Por cada día de retardo en la práctica se bajara 2 puntos. No se permite la entrega después de una semana de la fecha límite de la entrega.
- Enviar a la cuenta de correo un paquete zipeado que contendrá: un documento con los resultados de cada una de las actividades, así como los archivos fuentes (con extensión C).
- En el subject del correo escribir: **IP- practica 2- <nombrs completos>**.
- **CBI:** Los compañeros de CBI deben de entregar las actividades extras.

1 Objetivos

- 1) Utilizar la estructura de control condicional múltiple en el lenguaje C – **switch**. Especificando su utilización así como la distinción con el uso de sentencias **if** anidadas.
- 2) Utilizar la estructura de control iterativa **Mientras (While)** en el desarrollo de programas.
- 3) Utilizar la estructura de control iterativa **Hacer... Mientras (Do... While)** en el desarrollo de programas.
- 4) Utilizar la estructura de control iterativa **Para (For)** en el desarrollo de programas.
- 5) Determinar las diferencias entre el uso de las estructuras de control iterativa, y reportando cuales son los tipos de problemas que se pueden desarrollar con cada una de estas estructuras.

2 Introducción

En el desarrollo de esta practica se efectuara la fase de codificación ingresando en el IDE el código proporcionado para el desarrollo de ejercicios.

Para las actividades a desarrollar referentes a la estructuras de control iterativa, se efectuara la codificación de código proporcionado, así como la traducción de Pseudocódigo a Código en C, tanto el pseudocódigo como las definiciones de transformación son proporcionadas.

También se desarrollara problemas en donde se requiere que elabore cada una de las fases del modelo en cascada incluyendo las fases de análisis y diseño.

Ayuda complementaria

- Notas del Curso de Introducción a al Programación en el Capitulo 4.
- Guía Básica para el uso de C.
- Referencia de la bibliografía dentro de las notas del curso.

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

3 Desarrollo

La sección del desarrollo de la práctica se divide en una serie de actividades que permiten englobar, desarrollar y alcanzar cada uno de los objetivos de la práctica. Es necesario leer con atención cada una de las actividades a desarrollar.

Configurando el IDE Turbo C – Versión 2

1. Crear una carpeta con tus iniciales en la carpeta TCPROG.
Ejemplo: Las iniciales que se utilizaran son: GBP

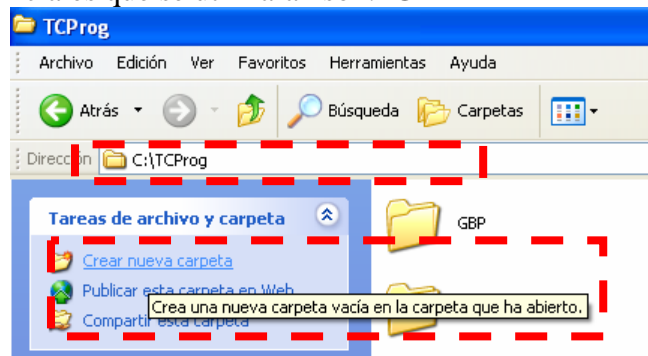


Figura 1.- Creación de carpeta en la ruta C:\TCPROG.

2. Ejecutar la aplicación TC.EXE la cual se encuentra en el directorio BIN/ en donde se instaló o descomprimió Turbo C (generalmente se encuentra en C:\TC\BIN). Al momento de ejecutarla se despliega la aplicación en un entorno de MS-DOS.

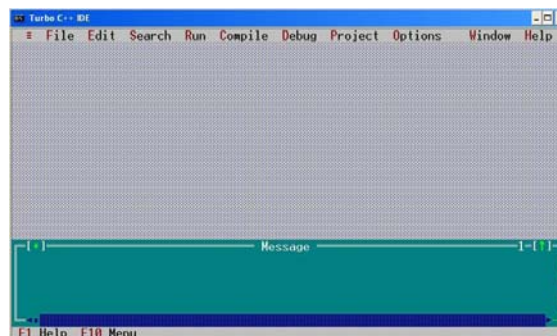


Figura 2.- Pantalla de inicio del IDE Turbo C.

3. Ajustar el directorio de salida de Turbo C en una sola carpeta, se selecciona del menú la siguiente ruta de opciones: **File** → **ChangeDir**.

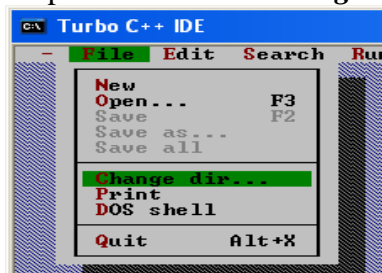


Figura 3.- Selección para cambiar la ruta de salida del IDE Turbo C.

4. En la opción de “*Directory Tree*” se busca la carpeta que se creó en la carpeta TCPCROG, indicando en donde se guardarán los archivos resultantes de la compilación con extensión **obj** y **exe**, así como los archivos creados con extensión **c**. Al finalizar se selecciona la opción **Chdir** y después **OK**.

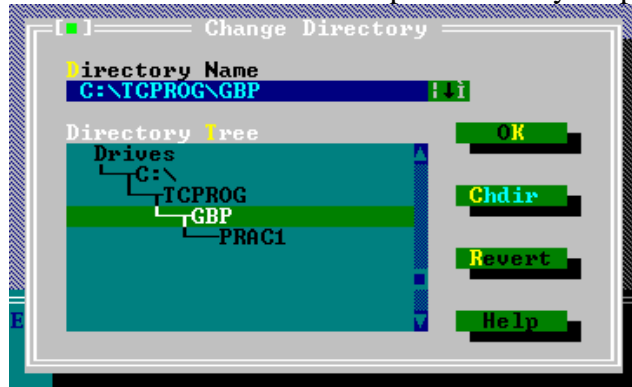


Figura 4.- Selección de la ruta de salida del IDE Turbo C.

Efectuada esta configuración todos los archivos que se generen estarán en la carpeta que se haya seleccionado.

3.1 Actividad 1 “Estructura de control condicional múltiple: If anidados Vs. Switch”

Con el desarrollo de esta actividad se alcanzará el objetivo 1 de la práctica.

La sentencia **switch** es una sentencia en C que es utilizada para seleccionar entre múltiples alternativas, principalmente cuando la selección se basa en el valor de una variable (o una expresión de control o selector). El valor de la expresión deberá ser de tipo carácter o entero (de tipo **char** o **int** en C, respectivamente), pero no del tipo flotante (**float** o **double**).

Definición 5: Selección condicional múltiple (switch)

Pseudocódigo	En C
<p>Si (<condición_1>) entonces <u>comienza</u> <instrucción A> <u>termina</u></p> <p>Otro <u>comienza</u> Si (<condición_2>) entonces <u>comienza</u> <instrucción B> <u>termina</u></p> <p>Otro <u>comienza</u> Si (<condición_3>) entonces <u>comienza</u> <instrucción C> <u>termina</u></p> <p>Otro <u>comienza</u> Si (<condición_4>) entonces</p>	<pre>//Variable puede ser char o int switch(<Contenido de Variable>) { case (<valor_1>) : { /*instrucción A */ break; //Se va a Z } case (<valor_2>) : { /*instrucción B */ break; //Se va a Z } case (<valor_3>) : { /*instrucción C */ break; //Se va a Z } case (<valor_4>) :</pre>

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

<pre> <u>comienza</u> <instrucción D> <u>termina</u> Otro <u>comienza</u> <instrucción E> <u>termina</u> <u>termina</u> <u>termina</u> <u>termina</u> </pre>	<pre> { /*instrucción D */ break; //Se va a Z } default : { /*instrucción E */ } //Se va a Z } /**instrucción Z**/ </pre>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

La sentencia **case** indica el ‘caso’ del resultado posible de la valor evaluado en el **switch**. Cuando ninguno de los **cases** se cumple, la sentencia **default** se ejecuta, es decir, ya no se considera el valor y se ejecuta cuando no se cumpla en ninguno de los **cases** anteriores.

La instrucción **break**, es utilizada en C para indicar la finalización de la sentencia del **case**, y del **switch**. Del ejemplo de código en C descrito en la tabla anterior, después de ejecutarse el las *instrucción X (A, B, C ó D)*, la instrucción **break** finaliza la acción y se pasa inmediatamente a ejecutar la instrucción Z (después del **switch**).

Problema

1. “Agenda – Switch”

Elabore un programa que pida el día de la semana como un número del 1 al 7, y muestre los mensajes siguientes dependiendo del día. Si se introduce un día distinto de 1 a 7 se marca un error.

- 1.- Lunes: Regresar al estudio, hacer tareas, entregar reportes, etc.
- 2.- Martes: Estudiar para la clase Introducción a la Programación.
- 3.- Miércoles: Hacer ejercicio e ir al Cine 2X1.
- 4.- Jueves: Estudiar para el examen.
- 5.- Viernes: Hacer el examen, y salir a ‘cenar’ con los amigos.
- 6.- Sábado: Ir a comprar cosas y después descansar todo el día.
- 7.- Domingo: Comer con la familia y a listarse para la siguiente semana.

Codificando

Entrada: Un entero indicando una de las posibles opciones a seleccionar.

Salida: Mensaje indicando que acciones deber de desarrollarse en la opción seleccionada.

Pseudocódigo “Agenda”

comienza

selección ← 0

Escribir (“Agenda de la semana”)

Escribir (“Seleccione una día de la semana indicando el numero correspondiente a la opción deseada.”)

Escribir (“1) Lunes 2) Martes 3) Miércoles 4) Jueves 5) Viernes
6) Sábado 7) Domingo ”)

Escribir (“Teclee su selección:”)

Leer (selección)

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

Si (selección = 1) entonces

com

Escribir (“Regresar al estudio, hacer tareas, entregar reportes, etc.”)

term

Otro

com

Si (selección = 2) entonces

com

Escribir (“Estudiar para la clase Introducción a la Programación.”)

term

Otro

com

Si (selección = 3) entonces

com

Escribir (“Hacer ejercicio e ir al Cine 2X1.”)

term

Otro

com

Si (selección = 4) entonces

com

Escribir (“Estudiar para el examen”)

term

Otro

com

Si (selección = 5) entonces

com

Escribir (“Hacer el examen, y salir a ‘cenar’ con los amigos”)

term

Otro

com

Si (selección = 6) entonces

com

Escribir (“Ir a comprar cosas y después descansar...”)

term

Otro

com

Si (selección = 7) entonces

com

Escribir (“Comer con la familia y a listarse...”)

term

term

term

term

term

term

term

termina

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

Actividad

Efectué la traducción del Pseudocódigo proporcionado considerando el uso de la estructura **switch**.

Puede apoyarse con el siguiente código el cual fue traducido utilizando sentencias **if-anidadas**.

Código C

```

/*
  Archivo: Agenda.c
  Descripción:
  Autor:
  Fecha:
*/

/*Incluir librerías utilizadas en el programas*/
#include <stdio.h>

int main()
{
    /*Declaración (e inicialización) de variables*/
    int seleccion = 0 ;

    printf ("\t Agenda de la semana\n\n");
    printf("Seleccione una día de la semana indicando el numero
           correspondiente a la opción deseada.\n");
    printf("\n\t1) Lunes \n\t2) Martes \n\t3) Mi,rcoles \n\t4) Jueves \n\t5)
           Viernes \n\t6) Sabado \n\t7) Domingo ");
    printf("\n\n\tTeclee su selección:");
    scanf("%d", &seleccion);

    printf("\n\tUsted debe de:\n\t ");
    if ( seleccion == 1 )
    {
        printf("\n\nRegresar al estudio, hacer tareas, entregar reportes, etc.")
            ;
    }
    else
    {
        if (seleccion == 2 )
        {
            printf("\n\nEstudiar para la clase Introducciøn a la Programaciøn.") ;
        }
        else
        {
            if (seleccion == 3 )
            {
                printf("\n\nHacer ejercicio e ir al Cine 2X1.") ;
            }
            else
            {
                if (seleccion == 4 )
                {
                    printf("\n\nEstudiar para el examen") ;
                }
                else
                {
                    if (seleccion == 5 )
                    {
                        printf("\n\nHacer el examen, y salir a 'cenar' con los
                               amigos");
                    }
                }
            }
        }
    }
}

```


Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

Puede apoyarse del siguiente código para efectuar la actividad.

Código C

```

/*
  Archivo: NumerosPositivos.c
  Descripción:
  Autor: */
/*Incluir librerías utilizadas en el programas*/
#include <stdio.h>

int main()
{
  /*Declaración (e inicialización) de variables*/
  int NEntero = 0 ;
  int NPositivos = 0 ;
  int Cont = 0;

  printf("\t Programa Números Positivos \n\n ");

  //Uso de Iteracion condicional - while (mientras)
  while ( Cont < 10)
  {
    printf(" Proporcione un número entero: " );
    scanf("%d", &NEntero);

    if(NEntero > 0 )
    {
      NPositivos = NPositivos + 1;
    }
    Cont = Cont + 1;
  } //fin while

  printf("El número de valores positivos tecleados es: %d", NPositivos);
  getch();
  return (0);
}

```

Responda las preguntas 4, 5 y 6 del cuestionario.

3.3 Actividad 3 “Estructura de Control Iterativa – While (Mientras) – Factorial de un Número”

Con el desarrollo de esta actividad se alcanzara el objetivo 2 de la práctica.

Problema:

Calcular el factorial de un número entero positivo .Considerar los casos iniciales factorial (0) = 1 y factorial (1) = 1.

Ejemplo: factorial (2) = 2*1, factorial (3) = 3*2*1 = 6, factorial (4) = 4*3*2*1 =12, etc.

Entrada: Número entero (numeroN).

Salida: Mensaje indicando el factorial de numeroN.

Procedimiento:

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

Pseudocódigo. “factorial”

comienza

Escribir (“Proporcione un número entero positivo:”)
 Leer (numeroN)
 factorial ← 1

mientras (numeroN > 0)

comienza

factorial ← factorial * numeroN
 numeroN ← numeroN - 1

finMientras

Escribir (factorial)

termina

Responda las preguntas 7, 8 y 9 del cuestionario.

3.4 Actividad 4 “Estructura de Control Iterativa – Do ... While (Hacer ... Mientras) –”

Con el desarrollo de esta actividad se alcanzará el objetivo 4 de la práctica.

La sentencia *do-while* presenta una similitud con el ciclo condicional – *while* (mientras), con la diferencia que **se ejecuta siempre al menos una vez**. La condición del ciclo *while* se evalúa al inicio del ciclo, mientras que la condición de *do-while* se evalúa al final. Es útil cuando se requiere que se desarrolle al menos una vez el cuerpo del ciclo.

Definición 7: Sentencia iterativa do... while (hacer ... Mientras)

Pseudocódigo	En C
hacer <u>comienza</u> <instrucciones A> <instrucciones N> termina mientras <condición_ciclo >	do { /**instrucciones A**/ /**instrucciones N**/ } while (< condición_ciclo >);

Actividad

Elaborar un programa que imprima los caracteres ASCII junto con el valor entero (entre el rango del 32 ‘ ’ al 126 ‘~’). Para cada escritura en pantalla, preguntar si se desea continuar. El usuario tecleara un de los caracteres de acuerdo a las decisión que tome: ‘s’ (‘S’) o ‘n’ (‘N’).

Para el desarrollo de esta actividad se puede hacer uso de la sección 4.6.3.2 de las notas del curso.

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

Pseudocódigo. “CaracteresASCII ”

comienza

caracter ← ‘ ‘

hacer

comienza

respuesta ← ‘N’

Escribir (caracter - caracter (entero))

caracter ← caracter + 1

Escribir (“Desea continuar S (s) o N (n):”)

Leer (respuesta)

termina

mientras ((respuesta = ‘S’ o respuesta ‘s’) Y caracter < 126)

termina

Responda las preguntas 7, 8 y 9 del cuestionario.

Trucos y Tips

- Al trabajar con Turbo C se trabaja en un entorno de MS-DOS, para poder regresar al entorno de Windows se teclaea: **Alt+tab** o bien la tecla de **windos**.
- Si desea tener a Turbo C corriendo como programa en una ventana en Windows se teclaea: **Alt + Enter**. Con esto se podrá intercambiar rápidamente de Windows a Turbo C o viceversa.
- El editor indica que el archivo a sufrido modificaciones colocando un asterisco blanco en la parte inferior izquierda, para guardar los cambios se presiona la tecla de atajo **F2** ó bien la siguiente secuencia de opciones: “**File** → **Save**”.

4 Resultados.

En esta sección se describen los resultados de cada una de las actividades, así como los problemas encontrados durante el desarrollo y el como se resolvieron cada uno de los problemas hasta llegar a completar la actividad.

4.1 Resultados actividad 1

<<En esta seccion se muestra los resultados de la actividad, bien de manera textual o grafica. Muchas veces es mejor considerar la forma grafica >>

4.2 Resultados actividad 2

4.3 Resultados actividad 3

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

4.4 Resultados actividad 4

Problemas encontrados a realizar la práctica

Id	Descripción	Solución
1	Error de sintaxis al codificar el programa.	Faltaba un punto y coma al finalizar la instrucción.

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

5 Cuestionario

Responda a las siguientes preguntas de acuerdo al desarrollo de la actividad

1. En la actividad 1, ¿En que casos es posible hacer uso de la sentencia *switch*?
2. En la actividad 1, ¿Cuál es la funcionalidad del caso *default* en la estructura *switch*?
3. En la actividad 1, ¿Cuál es la funcionalidad de las instrucciones *break* en la estructura *switch*?
4. En la actividad 2, ¿Comente qué modificaciones deberá desarrollarse en el código del programa si se desea contar el numero de ceros ingresados?
5. En el programa desarrollado en la actividad 2, si el contador lo inicia en 5, ¿Cuantas iteraciones desarrollará?
6. En el programa desarrollado en la actividad 2, si la condición la cambia a “Cont <= 20” ¿Cuantas iteraciones desarrollará?
7. En el programa de la actividad 3 y con la ayuda de una calculadora, indique ¿Cuál es el mayor número que es posible calcular el factorial?
8. En el programa de la actividad 3, modifique el tipo de dato de las variables *factorial* y *numeroN* de *int* a *unsigned int* e indique hasta que valor es posible calcular el valor del factorial.
9. De una breve explicación del porque la diferencia de resultados de las preguntas 7 y 8.
10. En el programa de la actividad 4, Investigue el código ASCII y determine si el programa manda a imprimir los caracteres de acuerdo a su valor en entero. En caso de que haya inconsistencias efectué las correcciones necesarias.
11. En el programa de la activada 4, existen tres formas de leer un carácter desde teclado, indique las diferencias entre ellas. Las funciones son *scanf()*, *getch()* y *getche()*.
12. En el programa de la activada 4, es posible hacer este programa utilizando la sentencia mientras?, Indique la forma en como lo desarrollaría.
13. En base a lo desarrollado en la actividad 2, 3 y 4. Indique ¿En qué problemas es posible utilizar una estructura de control **While**?, ¿En qué problemas es posible utilizar la estructura de control **Do.. While** y ¿En cuáles es posible utilizar ambas estructuras?

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

6 Conclusiones.

Presentar en esta sección las conclusiones de la práctica de acuerdo a los objetivos planteados y a los resultados obtenidos.

7 Referencias.

En caso de consultar algunos documentos, libros o paginas Web como referencia.

8 Actividades Extras

Las siguientes actividades sirven como reforzar y cubrir los objetivos de la práctica.

Ejercicio Extra 1 “*Agenda semanal con medio tiempo*”

Desarrolle un programa similar al de la actividad 1, considerando ahora que se seleccione a parte del día, si se desarrolla en la mañana o en la tarde.

Ejemplo: Se selecciona 1: lunes.

El programa debe de preguntar:

“Agenda del lunes por: 1) En la mañana 2) En la tarde”

El usuario teclea 1 y se despliega:

“El lunes por la mañana debe de ir a la escuela”

Ejercicio Extra 2 “*Serie de Fiboonacci - Mientras*”

Escribir la serie de Fibonacci especificando el número (entero positivo) de iteraciones a escribir. A continuación se especifica la regla para especificar la serie de Fibonacci.

- $f_i = f_{i-2} + f_{i-1}$
- $f_0 = 0$
- $f_1 = 1$

Ejemplo: Analizar las siguientes series de Fibonacci.

$f_2 = f_0 + f_1 = 0 + 1 = 1$, $f_3 = f_1 + f_2 = 1 + 1 = 2$, $f_4 = 3$, $f_5 = 5$, $f_6 = 8$, ...

0, 1, 1, 2, 3, 5, 8, 13, ...

Ejercicio Extra 3 “*Contador con respuesta – Hacer.. Mientras*”

Elabore un programa que utilice un contador (inicializado en 0) para imprimir su contenido cada vez que el usuario teclea la letra S. Ejemplo:

```
0
Desea continuar (S/N): S
1
Desea continuar (S/N): S
2
Desea continuar (S/N): S
3
Desea continuar (S/N): N
Hasta pronto!!!
```

Sugerencia: El contador no es considerado dentro de la condición, sin embargo la letra que tecleo el usuario ‘S’ o ‘N’, es importante para condición.

Universidad Autónoma Metropolitana – Iztapalapa	Trimestre: 08-I
Curso de Introducción a la Programación – Practica 1	Práctica 2

Ejercicio Extra 4 “Agenda en un Ciclo – Hacer.. Mientras”

Desarrolle un programa similar al de agenda, considerando que el usuario tenga la opción de terminar de desplegar la información de su agenda con la opción 'N' o 'n'.

Ejercicio Extra 5 “Ciclos Infinitos”

Desarrolle al menos cinco formas de desarrollar de efectuar ciclos infinitos con las estructuras de control *While* y *Do... While*. Haga uso de la estructura *switch* para efectuar la selección de cualquiera de las cinco formas.

Sugerencia: Es necesario investigar con anticipación la forma de terminar un programa que se encuentre en un ciclo infinito.