

Práctica 2. Estructuras de control selectivas

24-25 de Febrero de 2016

Objetivos de la práctica:

- Adquirir la capacidad de diseñar algoritmos para la resolución de problemas básicos.
- Conocer cómo se traduce un algoritmo escrito en pseudocódigo en un lenguaje de programación.

1. Introducción

En la práctica anterior vimos qué etapas teníamos que seguir cuando programamos en lenguaje C:

- En papel, diseñar el algoritmo que resuelve el problema que queremos resolver.
- Traducir el algoritmo a lenguaje C. Para ello usaremos un editor de textos como por ejemplo el *gedit*.
- Compilar el programa para obtener un ejecutable. Vamos a utilizar el compilador *gcc*. El comando a utilizar sería: `$gcc -o nombreprograma nombreprograma.c`
- Ejecutar el programa y probar que su funcionamiento es el deseado. Emplearemos el comando `$/nombreprograma`

2. Estructuras de control selectivas

Nos permiten tomar decisiones. Hay varios tipos:

- Alternativa simple
- Alternativa doble
- Alternativa múltiple

2.1. Alternativa simple

Si se cumple una condición, se ejecutan unas instrucciones. Si no, no se ejecutan.

```
if (condición){
    //instrucciones
}
//instrucciones
```

2.2. Alternativa doble

Si se cumple una condición, se ejecutan unas instrucciones. Si no se cumple, se ejecutan otras instrucciones.

```
if (condición){
    //instrucciones
}
else{
    //instrucciones
}
//instrucciones
```

2.3. Alternativa múltiple

En función del valor de una condición se ejecutan distintos tipos de instrucciones. En C se puede hacer de dos formas, con if-else anidados:

```
if (condición){
    //instrucciones
}
else if(condición){
    //instrucciones
} else if(condición){
    //instrucciones
} else if(condición){
    //instrucciones
} else {
    //instrucciones
}
//instrucciones
```

O bien usando la estructura switch:

```
switch (variable){
    case 1:
        //instrucciones si variable vale 1
        break;
    case 2:
        //instrucciones si variable vale 2
        break;
    ...
    default:
        //instrucciones si variable no vale nada de lo anterior
}
```

```
//instrucciones
```

2.4. Condiciones

Para evaluar una condición, podemos usar los operadores de comparación:

- Mayor o igual que ($a \geq b$)
- Mayor que ($a > b$)
- Igual que ($a == b$)
- Distinto que ($a != b$)
- Menor o igual que ($a \leq b$)
- Menor que ($a < b$)
- Que se cumpla una condición y otra condición (condición1) && (condición2)
- Que se cumpla una condición u otra condición (condición1) || (condición2)
- Que no se cumpla una condición !(condición)

3. Algoritmos

3.1. Algoritmo Detecta_tres

El algoritmo siguiente detecta si el número introducido por teclado es el 3:

ALGORITMO Detecta_tres

ENTRADAS:

Num: Entero ; Número que se lee

SALIDAS:

VARIABLES:

Num: Entero

INICIO

ESCRIBA "Escribe un número: "

LEA Num

SI Num == 3 ENTONCES

 ESCRIBA "El número es el 3"

SINO

 ESCRIBA "El número no es el 3"

FINSI

ESCRIBA "Fin del algoritmo"

FIN

3.2. Algoritmo Calcula_nota

El algoritmo siguiente convierte una nota numérica en su correspondiente calificación:

```
ALGORITMO Calcula_nota
  ENTRADAS:
    Num: Entero ; Nota numérica
  SALIDAS:
  VARIABLES:
    Num: Entero ; Nota numérica
  INICIO
    ESCRIBA "Escribe la nota numérica: "
    LEA Num
    SI Num == 5 O Num == 6 ENTONCES
      ESCRIBA "Aprobado"
    SINO SI Num == 7 O Num == 8 ENTONCES
      ESCRIBA "Notable"
      SINO SI Num == 9 O Num == 10 ENTONCES
        ESCRIBA "Sobresaliente"
      SINO
        ESCRIBA "Suspenso"
      FINSI
    FINSI
  FINSI
FIN
```

3.3. Algoritmo Calcula_nota_v2

El algoritmo siguiente convierte una nota numérica en su correspondiente calificación:

```
ALGORITMO Calcula_nota_v2
  ENTRADAS:
    Num: Entero ; Nota numérica
  SALIDAS:
  VARIABLES:
    Num: Entero ; Nota numérica
  INICIO
    ESCRIBA "Escribe la nota numérica: "
    LEA Num
    SEGÚN Num HACER
      CASO 5: ESCRIBA "Aprobado"
      CASO 6: ESCRIBA "Aprobado"
      CASO 7: ESCRIBA "Notable"
      CASO 8: ESCRIBA "Notable"
      CASO 9: ESCRIBA "Sobresaliente"
      CASO 10: ESCRIBA "Sobresaliente"
    SINO
      ESCRIBA "Suspenso"
    FINSEGUN
  FIN
```

4. Programación en lenguaje C

Los programas escritos en el lenguaje C se guardan en archivos con extensión .c. En esta práctica nuestros programas van a tener la siguiente estructura básica:

```
//Librería que contiene las funciones scanf y printf
#include <stdio.h>

//Función principal del programa
int main ()
{

    //Aquí van las instrucciones de nuestro programa.
    //Todas acaban en ;

    return 0;
}
```

4.1. Programa Detecta_tres

Si traducimos el algoritmo Detecta_tres a lenguaje C nos quedaría el programa siguiente:

```
//Librería que contiene las funciones scanf y printf
#include <stdio.h>

//Función principal del programa
int main ()
{
    // Este programa indica si el número leído por teclado es 3

    // Declaro las variables de mi función
    int Num;

    //Sustituyo la función ESCRIBA "cadena" por printf
    printf("Escribe un número: ");

    //Sustituyo la función LEA Num por scanf ("%d", &variableEntera);
    scanf("%d", &Num); //Guarda el número leído en la variable Num

    //Compruebo si el número introducido es el 3
    if (Num == 3){
        //Sustituyo la función ESCRIBA "cadena" por printf
        printf("El número introducido es el 3\n");
    }
    else {
        //Sustituyo la función ESCRIBA "cadena" por printf
        printf("El número introducido no es el 3\n");
    }
}
```

```
//Sustituyo la función ESCRIBA "cadena" por printf
printf("Fin del algoritmo\n");

//Fin del programa
return 0;
}
```

4.2. Programa Calcula_notas

La traducción del algoritmo Calcula_notas a lenguaje C sería la siguiente:

```
//Librería que contiene las funciones scanf y printf
#include <stdio.h>

//Función principal del programa
int main ()
{
    // Este programa convierte una nota numérica a su
    // calificación correspondiente

    // Declaro las variables de mi función
    int Num;

    //Sustituyo la función ESCRIBA "cadena" por printf
    printf("Escribe la nota numérica: ");

    //Sustituyo la función LEA Num por scanf ("%d", &variableEntera);
    scanf("%d", &Num); //Guarda el número leído en la variable Num

    //Compruebo qué nota es para imprimir la cadena correspondiente
    if ((Num == 5) || (Num == 6)){
        //Sustituyo la función ESCRIBA "cadena" por printf
        printf("Aprobado");
    }
    else{
        if ((Num == 7) || (Num == 8)){
            //Sustituyo la función ESCRIBA "cadena" por printf
            printf("Notable");
        }
        else{
            if ((Num == 9) || (Num == 10)){
                //Sustituyo la función ESCRIBA "cadena" por printf
                printf("Sobresaliente");
            }
            else{
                //Sustituyo la función ESCRIBA "cadena" por printf
                printf("Suspenso");
            }
        }
    }
}
```

```
}

//Fin del programa
return 0;
}
```

4.3. Programa Calcula_notas_v2

La traducción del algoritmo Calcula_notas_v2 a lenguaje C sería la siguiente:

```
//Librería que contiene las funciones scanf y printf
#include <stdio.h>

//Función principal del programa
int main ()
{
    // Este programa convierte una nota numérica a su
    // calificación correspondiente

    // Declaro las variables de mi función
    int Num;

    //Sustituyo la función ESCRIBA "cadena" por printf
    printf("Escribe la nota numérica: ");

    //Sustituyo la función LEA Num por scanf ("%d", &variableEntera);
    scanf("%d", &Num); //Guarda el número leído en la variable Num

    //Compruebo qué nota es para imprimir la cadena correspondiente
    switch (Num){
        //Sustituyo la función ESCRIBA "cadena" por printf

        //Imprimo lo mismo si la nota es 5 o si es 6
        case 5:
        case 6: printf("Aprobado\n");
                break; //Sale del switch porque ya entró en un caso

        //Imprimo lo mismo si la nota es 7 o si es 8
        case 7:
        case 8: printf("Notable\n");
                break; //Sale del switch porque ya entró en un caso

        //Imprimo lo mismo si la nota es 9 o si es 10
        case 9:
        case 10: printf("Sobresaliente\n");
                 break; //Sale del switch porque ya entró en un caso
        //Si no es ningún valor de los anteriores, Suspenso
        default:
                printf("Suspenso\n");
    }
```

```
    }

    //Fin del programa
    return 0;
}
```

5. Ejercicios propuestos

5.1. Ejercicio 1

- Escribe en un archivo con extensión `.c` el programa `Detecta_tres`. Compíllalo con el compilador `gcc`. Después, ejecútalo para comprobar su funcionamiento.

5.2. Ejercicio 2

- Escribe en un archivo con extensión `.c` el programa `Calcula_nota`. Compíllalo con el compilador `gcc`. Después, ejecútalo para comprobar su funcionamiento.

5.3. Ejercicio 3

- Escribe en un archivo con extensión `.c` el programa `Calcula_nota_v2`. Compíllalo con el compilador `gcc`. Después, ejecútalo para comprobar su funcionamiento.

5.4. Ejercicio 4

- Crea un programa que lea un número y escriba el mes al que corresponde imprimiendo su nombre por pantalla. Si es el 1, habrá que imprimir Enero, si es el 2, Febrero, etc. Si no es un número del 1 al 12 debe imprimir un mensaje de error.