

# Fundamentos de Programación

## Grado en Ingeniería Informática

### Relación de Ejercicios IV-a

Profesor: David A. Pelta

1. En este ejercicio se deben diseñar una serie de filtros para la entrada de datos. Implementélos en un único programa.
  - a) un filtro para “letras”.
  - b) un filtro para números positivos pares.
  - c) un filtro que lea dos números  $n1, n2$ , controlando que  $n1 < n2$ .
  - d) un filtro para números enteros terminados en un dígito impar.
2. Descargue desde PRADO el programa `nroEnDigitosII.cpp` que permite descomponer un número entero  $X$  en sus dígitos. Estúdielo. Luego implemente las tareas necesarias para resolver los siguientes problemas:
  - a) calcular la suma de los dígitos de  $X$ .
  - b) calcular el número de dígitos pares de  $X$ .
  - c) calcular la suma de los dígitos impares de  $X$ .
3. Implemente bucles `while` que permitan obtener las siguientes salidas:
  - a)  $10, 9, 8, 7, \dots, 1$
  - b)  $-10, -8, \dots, 8, 10$
  - c)  $(10, 0), (9, 1), (8, 2), \dots, (0, 10)$
4. A partir de un capital inicial  $C$ , contratamos un plazo fijo anual con un interés del  $x\%$ . Al final del primer año, con el capital inicial más los intereses obtenidos, renovamos el plazo fijo con el mismo interés. Diseñe e implemente un programa que permita calcular cuantos años pasarán hasta doblar el capital inicial.
5. Descargue desde PRADO el programa `promedio.cpp` que lee una serie de valores enteros positivos (hasta que se lee -1) y calcula el promedio de los mismos. Modifique ese código para resolver los siguientes problemas:
  - a) calcular la diferencia entre el promedio de los números impares y el promedio de los pares.

- b) obtener el máximo y el mínimo de los valores leídos.
  - c) informar si los datos leídos forman una secuencia creciente (el valor actual debe compararse con el anterior).
  - d) informar la longitud de la mayor secuencia creciente. Por ejemplo, si la entrada es 5,6,3,1,2,3,4,5,0,1,2 se puede ver que hay tres secuencias crecientes ( $\{5,6\}$ ),  $\{1,2,3,4,5\}$ ,  $\{0,1,2\}$ ). Por tanto, la mayor secuencia creciente tiene longitud 5.
6. Se pide diseñar un programa para jugar a adivinar un número entre 1 y 100. El juego tiene que dar pistas de si el número introducido por el jugador está por encima o por debajo del número introducido. Como reglas de parada se consideran los siguientes dos casos: a) se ha acertado b) se decide abandonar el juego (decida cómo quiere especificar esta opción) Para poder generar números aleatorios en un rango determinado será necesario incluir las instrucciones que aparecen en el código de la Figura 1.

La sentencia `srand(time(&tiempo))` debe ejecutarse una única vez al principio del programa y sirve para inicializar la secuencia de números aleatorios. Posteriormente, cada vez que se ejecute la sentencia `incognita = (rand() % NUM_VALORES) + MIN`; se obtendrá un valor aleatorio (pseudoaleatorio). Realizar el mismo ejercicio pero permitiendo jugar tantas veces como lo desee el jugador.

```
#include <iostream>
#include <ctime>
#include <cstdlib>
using namespace std;
int main(){
    const int MIN = 1, MAX = 100;
    const NUM_VALORES = MAX-MIN + 1; // rango
    int incognita; // número generado
    time_t tiempo;
    // Inicialización de la secuencia:
    srand(time(&tiempo));
    // Generación de un número aleatorio incognita:
    // MIN <= incognita <= MAX
    incognita = (rand() % NUM_VALORES) + MIN;
```

Figura 1: Código para generar números enteros aleatorios entre MIN y MAX.

7. Una empresa que tiene tres sucursales decide llevar la contabilidad de las ventas de sus productos a lo largo de una semana. Para ello registra cada venta con tres valores: el identificador de la sucursal (1, 2 o 3), el código del producto codificado como un carácter (a, b ó c) y el número de unidades vendidas. Se pide diseñar un programa que lea desde el teclado una serie de registros compuestos por sucursal, producto, y unidades vendidas y diga cuál es la sucursal que más productos ha vendido. La serie de datos termina cuando la sucursal introducida vale -1. Por ejemplo, con la serie de datos

2	a	20
1	b	10
1	b	4
3	c	40
1	a	1
2	b	15
1	a	1
1	c	2
2	b	6
-1		

la sucursal que más productos ha vendido es la número 2 con 41 unidades totales. Para comprobar que el programa funciona correctamente, cread un fichero de texto y re-dirigid la entrada a dicho fichero