

# ***Prog. Básica - Laboratorio 5 Subprogramas***

**Nota:** Por norma general, en los programas de pruebas que se facilitan hay que añadir más casos de prueba porque no se incluyen todos necesariamente. Además, muchos subprogramas están sin especificar (precondición, postcondición), y en esos casos deberéis escribir la especificación vosotros mismos.

## **1º ejercicio**

**Decimal a binario:** Escribir un subprograma que dado un número entero en base decimal calcule su equivalente binario. Utilizar las plantillas *decimal\_a\_binario.adb* y *prueba\_decimal\_a\_binario.adb* para ADA y *decimal\_a\_binario.py* para PYTHON.

Para pasar un número de decimal a binario iremos dividiendo el número decimal entre 2 y quedándonos con los restos.

Ejemplo: 4 resto 2=0    2 resto 2=0    1 resto 2=1, y el equivalente en binario es 100. Es decir  $1*(10**2)+0*(10**1)+0*(10**0)=100$

## **2º ejercicio**

**Ordenar tres números:** Implementar un subprograma que dados tres números enteros positivos los ordene de mayor a menor. Utilizar las plantillas *intercambiar.adb*, *ordenar\_dos\_numeros.adb*, *ordenar\_tres\_numeros.adb* y *prueba\_ordenar\_tres\_numeros.adb* para ADA y *ordenar\_tres\_numeros.py* para PYTHON.

## **3º ejercicio**

**Primer número primo y capicúa:** Escribir un subprograma que dado un entero positivo obtenga el primer número primo y capicúa mayor que él. Utilizar las plantillas *es\_primo.adb*, *prueba\_es\_primo.adb*, *es\_capicua.adb*, *prueba\_es\_capicua.adb*, *primo\_y\_capicua\_mayor.adb* y *prueba\_primo\_y\_capicua\_mayor.adb* para ADA y *prueba\_primo\_y\_capicua\_mayor.py* para PYTHON.

**NOTA:** De cara a la eficiencia, no conviene llamar a ambos subprogramas (*es\_primo* y *es\_capicua*) a la vez, dado que cuando el primero al que se haya llamado devuelve *false* no tiene sentido seguir comprobando nada más.

## **4º ejercicio**

**Suma de dos números binarios.** Escribir un subprograma que dados dos números binarios obtenga el resultado de la suma de ambos. Utilizar las plantillas *suma\_binaria.adb* y *prueba\_suma\_binaria.adb* para ADA y *suma\_binaria.py* para PYTHON.

**NOTA:** Una solución de este ejercicio basada en la reutilización de código consistiría en convertir los dos sumandos a base decimal, después realizar la suma y por último convertir el resultado a binario. Sin embargo, en este problema NO se acepta dicha solución por ineficiente. En su lugar se pide realizar la suma directamente en binario, sumando dígito a dígito y controlando el valor de las llevadas.

## **5º ejercicio**

**Números de teléfono.** Escribir el subprograma *coste\_de\_llamada* que dados el número de teléfono al que se ha realizado la llamada, los minutos hablados y dos booleanos (que indicarán, respectivamente, si se dispone de tarifa plana a móviles y si se dispone de tarifa plana a fijos) calcule el precio de la llamada y el tipo de número de teléfono al que se ha llamado.

Dado que las tarificaciones varían dependiendo, por ejemplo, de si se ha llamado a un número fijo, un móvil, un número de cobro revertido, de tarificación especial etc, lo primero que se deberá hacer será identificar el tipo de llamada que se ha realizado. A modo de resumen, la siguiente tabla describe las características de los tipos de número de teléfono:

**NOTA:** Como precondición, se considerará para este ejercicio que los números de teléfono contendrán 5 ó 9 dígitos. No se tendrá en cuenta ningún número que no cumpla esta precondición.

Tipo	Número de dígitos	Características	Tarificación
1 (número corto)	5 dígitos	Contiene 5 dígitos	Gratuita
2 (móvil)	9 dígitos	Comienzan por 6 ó 7	0,02 euros/min, salvo si el usuario dispone de tarifa plana a móviles. En ese caso es Gratuita
3 (fijo de cobro revertido)	9 dígitos	Comienzan por 800 ó 900	Gratuita
4 (fijo coste semi-ampliado)	9 dígitos	Comienzan por 902	0,025 euros/min
5 (fijo de coste compartido)	9 dígitos	Comienzan por 901	0,012 euros/min
6 (fijo de coste ampliado)	9 dígitos	Comienzan por 803, 806, 807, 905 ó 907	0,3 euros/min
7 (fijo estándar locales)	9 dígitos	Comienzan por 94 (solo serán de Euskadi y Navarra, así que pueden ser 945 Araba, 943 Gipuzkoa, 948 Navarra ó 94 Bizkaia)	0,004 euros/min, salvo si el usuario dispone de tarifa plana a fijos. En ese caso es Gratuita
8 (fijo estándar no local)	9 dígitos	Resto de números (incluyendo los de las provincias 941, 942, 947 y 949, que comenzando por 94 no son de Bizkaia)	0,01 euros/min

Por ejemplo, una llamada al número 944119090 sería de tipo 7 (fijo estándar local) porque dicho número pertenece a Euskadi, dado que comienza por 94 pero no por 941, 942, 947 ni 949. Por su parte, una llamada al número 941020202 sería de tipo 8 (fijo estándar no local) porque no pertenece a Euskadi ni Navarra.

Para implementar el subprograma *coste\_de\_llamada*, se implementarán o se hará uso de los siguientes subprogramas:

● *tipo\_de\_numero* que dado un número de teléfono devolverá un entero que representa el tipo del teléfono según se ha descrito en la tabla anterior. Para poder identificar el tipo de número de teléfono, habrá que implementar y emplear los siguientes subprogramas:

- *numero\_de\_digitos()*, que dado un número devolverá cuántos dígitos tiene.
- *es\_numero\_corto()*, que dado un número de teléfono dirá si tiene exactamente 5 dígitos o no.
- *es\_un\_movil()* que dado un número de teléfono devolverá un booleano que indica si es un móvil (es decir, comienza por 6 ó 7).
- *es\_de\_cobro\_revertido()*, que dado un número de teléfono devolverá un booleano que indica si el número comienza por 800 ó 900.
- *es\_de\_coste\_semi\_ampliado()*, que dado un número de teléfono devolverá un booleano que indica si el número comienza por 902.
- *es\_de\_coste\_compartido()*, que dado un número de teléfono devolverá un booleano que indica si el teléfono comienza por 901.
- *es\_de\_coste\_ampliado()*, que dado un número de teléfono devolverá un booleano que indica si el número comienza por alguno de los siguiente códigos: 803, 806, 807, 905 ó 907.
- *es\_fijo\_local()*, que dado un número de teléfono devolverá un booleano que indica si el número comienza por alguno de los siguiente códigos: 943, 945, 948, ó 94 (exceptuando 941, 942, 947, y 949, que comienzan por 94 pero no pertenecen a Bizkaia).

● *coste\_de\_llamada\_por\_tipo*, que dados el tipo de número de teléfono y la duración en minutos devolverá el coste de la llamada.

Utilizar las plantillas *coste\_de\_llamada.adb*, *prueba\_coste\_de\_llamada.adb*, *tipo\_de\_numero.adb*, *coste\_de\_llamada\_por\_tipo.adb*, *numero\_de\_digitos.adb*, *es\_de\_cobro\_revertido.adb*, *es\_de\_coste\_ampliado.adb*, *es\_de\_coste\_compartido.adb*, *es\_de\_coste\_semi\_ampliado.adb*, *es\_fijo\_local.adb*, *es\_numero\_corto.adb*, y *es\_un\_movil.adb* para ADA y *prueba\_coste\_de\_llamada.py* para PYTHON.

Junto con las plantillas están disponibles los ficheros de apoyo siguientes, que se podrán utilizar tal cual se proporcionan (esto es, sin necesidad de modificarlos): *prueba\_es\_de\_cobro\_revertido.adb*, *prueba\_es\_de\_coste\_ampliado.adb*, *prueba\_es\_de\_coste\_compartido.adb*, *prueba\_es\_de\_coste\_semi\_ampliado.adb*, *prueba\_es\_fijo\_local.adb*, *prueba\_es\_numero\_corto.adb* y *prueba\_es\_un\_movil.adb*. La plantilla que necesariamente habrá que completar con los casos de prueba que faltan es *prueba\_coste\_de\_llamada.adb*.

## 6º ejercicio

**Número medio:** Escribir un subprograma que dado un entero positivo diga si es un número medio o no. Un número es medio si la suma de los números positivos inferiores a él es igual a la suma de números consecutivos superiores a él.

Ejemplos:

6 es número medio porque

la suma de los números inferiores a 6 es

$$1+2+3+4+5=15$$

una suma de números consecutivos superiores a 6 es

$$7+8=15$$

7 no es número medio porque

la suma de los números inferiores a 7 es

$$1+2+3+4+5+6=21$$

ninguna suma de números consecutivos superiores a 7 vale lo mismo que la suma de números inferiores.

$$8+9=17$$

$$8+9+10=27...$$

(Llegados a este punto ya sabemos que no vamos a encontrar ninguna suma de números superiores a 7 tal que el resultado sea igual a 21)

35 es número medio porque

la suma de los números inferiores a 35 es

$$1+2+3+4+5+...34=595$$

una suma de números consecutivos superiores a 35 es

$$36+37+...+49=595$$

Para este ejercicio no se facilitan plantillas, por lo que deberéis resolver el problema y realizar las pruebas desde cero.