

Parcial del Proyecto IV - Tema B

Ejercicio 1 - [assert() y asignaciones múltiples]

Ejercicio 1: Siendo D_1 y D_2 los dos últimos números de tu DNI (XX.XXX.XD₁D₂), traducir el siguiente programa en formalismo básico al lenguaje C:

```
var x, y, z, n, i : Int;
[[σ0 : (x ↦ 3, y ↦ 7, z ↦ 1, n ↦ 0, i ↦ 0)]]
{Pre: x = X, y = Y, x mod z = 0, y mod z = 0}
x := x div z;
y := y * z;
n := 2 * (D1 + D2) + 1;
i := 0
do (i < n) →
    x, y, i := y div z, x * z, i + 1
od
{Post: x = Y, y = X}
```

Asegurarse que la traducción preserve el funcionamiento de la asignación múltiple efectuada dentro del ciclo.

Los valores iniciales de las variables x , y , z deben obtenerse del usuario usando la función `pedirEntero()`. Al finalizar el programa se deben mostrar los valores de las variables x , y usando la función `imprimeEntero()`. Se deben verificar las pre y post condiciones usando la función `assert()`.

Ejercicio 2 - [funciones y struct]

Ejercicio 2: Dada la estructura:

```
typedef struct {
    int altura;           // Altura en centímetros
    int nacimiento;     // Año de nacimiento
    int titulos;        // Cantidad de torneos ganados
    bool usa_izquierda; // True si es zurdo / zurda
} tenista;
```

a) Elegir dos variables del tipo `tenista` de entre:

`roger_federer`, `rafael_nadal`, `serena_williams`, `maria_sharapova`,
y escribir una función `main()` donde se les asigne la información correspondiente
para cada componente a esas dos variables. La función tendría la siguiente forma:

```
int main(void) {
    tenista roger_federer, serena_williams;
    /* A partir de aquí asignar valores a los componentes */
    /* de las dos variables elegidas */
    return 0;
}
```

Deben basarse en los siguientes datos:

- **Roger Federer** es un tenista profesional nacido en Suiza en el año 1981. Con una altura de 1.85 metros (185cm) y siendo diestro ha ganado hasta el día de la fecha 103 torneos profesionales.
- **Rafael Nadal** es un tenista español nacido el 3 de junio del 86. Su altura es de 6' 1" (185cm). A lo largo de su carrera ha ganado 85 títulos, siendo después de Jimmy Connors el jugador zurdo más ganador de todos los tiempos.
- **Serena Williams** es una tenista profesional nacida en Estados Unidos en el año 1981. En su carrera ha ganado 73 títulos. Su mano hábil es la derecha y aunque su altura es de 1.75 metros (175cm) tiene uno de los saques más fuertes de todo el circuito WTA.
- **María Sharapova** es una tenista Rusa nacida el 19 de abril del 87. Su altura es de 6' 2" (188cm) y es diestra. A lo largo de su carrera ha ganado 36 títulos y es considerada una de las mejores jugadoras de su época.

b) Definir la función

```
bool tiene_mas_altura(tenista t0, tenista t1);
```

que devuelve `true` si `t0` tiene mayor estatura que `t1`. Luego en la misma función `main()` del apartado a) usar la función recién definida con las dos variables de tipo `tenista` elegidas y mostrar el resultado por pantalla.

Ejercicio 3 - [arreglos]

Ejercicio 3: Programar en C la función:

```
int multiplica_pares(int a[], int tam);
```

que calcula el producto de todos los números pares del arreglo `a`. Escribir además una función `main()` donde se le pida al usuario que ingrese los valores del arreglo usando la función `pedirArreglo()`. Finalmente mostrar por pantalla el resultado de la aplicación de `multiplica_pares()` al arreglo ingresado por el usuario. El tamaño del arreglo pueden asumirlo constante (definiendo una constante `N` que indique la cantidad de elementos) o permitirle al usuario que elija una longitud.

Ejercicio 4* - [Números perfectos]

Ejercicio 4*: Un número perfecto es un entero positivo que es igual a la suma de sus divisores propios positivos. Por ejemplo, 6 es un número perfecto porque sus divisores propios son 1, 2 y 3; y $6 = 1 + 2 + 3$. Notar que 6 es divisor de 6, pero **no es un divisor propio** pues no vale usar el mismo número como divisor. Otros ejemplos: 28, 496 y 8128.

$$28 = 1 + 2 + 4 + 7 + 14$$

$$496 = 1 + 2 + 4 + 8 + 16 + 31 + 62 + 124 + 248$$

$$8128 = 1 + 2 + 4 + 8 + 16 + 32 + 64 + 127 + 254 + 508 + 1016 + 2032 + 4064$$

Programar en lenguaje C las funciones:

```
bool es_perfecto(int n);
```

que devuelve true si el número n es perfecto, y

```
int kesimo_perfecto(int k);
```

que devuelve el k-ésimo número perfecto.

Escribir una función `main()` que solicite al usuario ingresar un entero `k`. Si es negativo imprimir un mensaje de error, y si es no negativo imprimir el resultado devuelto por la función `kesimo_perfecto()` para ese entero `k`.

ADVERTENCIA: Probar el programa con valores pequeños para `k` ya que es difícil encontrar números perfectos y podría llevar mucho tiempo de búsqueda (pensar que el quinto número perfecto tiene ocho cifras!)