

Examen tipo A

Ejercicio 1 (3 puntos) El logaritmo natural de un número positivo x se puede aproximar con los n primeros términos de la siguiente serie:

$$\ln x = \sum_{i=0}^{\infty} \frac{1}{2i+1} \left(\frac{x^2-1}{x^2+1} \right)^{2i+1}$$

- Escribe una **función recursiva** que, dados un número real positivo x y un número natural n , calcule la suma de los n primeros términos de la serie. Se puede utilizar la función `pow(double base, double exponente)` de la librería `cmath`.
- Escribe una función (no recursiva) que, para un número dado x , calcule cuántos términos de la serie son necesarios para que el error del valor devuelto por la función del apartado a) sea menor a 10^{-2} respecto al valor proporcionado por la función `log(double x)` de la librería `cmath`. Utiliza la función que compara números reales de forma absoluta vista en clase.

Ejercicio 2 (1 punto) Se dispone de la función definida a continuación. Nos dicen que esta función tiene en algunos casos un comportamiento incorrecto. Determina razonadamente cuál es ese comportamiento incorrecto y en qué casos se produce.

```
double f(double x, int a) {
    if (a % 2 == 0)
        return f(x*x, a/2);
    else if (a % 2 != 0)
        return x*f(x, a-1);
    else if (a == 0)
        return 1;
}
```

Ejercicio 3 (3 puntos) Dos números naturales x e y son **amigos** si la suma de los divisores propios de x (excluido x) es igual a y y viceversa.

- Escribe una función booleana que determine si dos números dados son amigos. Debe utilizarse una función auxiliar que calcule la suma de los divisores de un número.
- Escribe un procedimiento que pida dos números por la pantalla y los lea del teclado.
- Escribe una función `main()` que lea pares de números positivos con el procedimiento definido en b) y determine si son amigos con la función definida en a) hasta que se introduzcan dos ceros por el teclado. Cuando se tecleen dos ceros, el programa debe terminar **sin llamar** a la función definida en a).

Ejercicio 4 (2 puntos) Escribe una función que calcule la integral definida de la función $f(x) = \cos^2(x)$ en un intervalo $[a, b]$. Debe recibir un parámetro implícito que indique el intervalo mínimo a considerar (por defecto 10^{-4}).

Ejercicio 5 (1 punto) Escribe qué resultado producirá en la pantalla el siguiente programa:

```
#include <iostream>
using namespace std;

int incr(int q, int& p) {
    static int r=0;
    r = q+p; p = r;
    return r;
}

int main() {
    int w = 4;
    for (int r=10; r>0; r -= 2)
        cout << r << " " << incr(r,w) << endl;
    return 0;
}
```

Examen tipo B

Ejercicio 1 (3 puntos) El coseno de un número x se puede aproximar con los n primeros términos de la siguiente serie:

$$\cos x = \sum_{i=0}^{\infty} \frac{(-1)^i}{(2i)!} x^{2i}$$

- Escribe una **función recursiva** que, dados un número real x y un número natural n , calcule la suma de los n primeros términos de la serie. Se puede utilizar la función `pow(double base, double exponente)` de la librería `cmath`. El cálculo del factorial hazlo con otra función recursiva.
- Escribe una función (no recursiva) que, para un número dado x , calcule cuántos términos de la serie son necesarios para que el error del valor devuelto por la función del apartado a) sea menor a 10^{-2} respecto al valor proporcionado por la función `cos(double x)` de la librería `cmath`. Utiliza la función que compara números reales de forma absoluta vista en clase.

Ejercicio 2 (1 punto) Se dispone de la función definida a continuación. Nos dicen que esta función tiene en algunos casos un comportamiento incorrecto. Determina razonadamente cuál es ese comportamiento incorrecto y en qué casos se produce.

```
double f(double x, int a) {  
    if (a == 0)  
        return 1;  
    else  
        return x*x*f(x, a-2);  
}
```

Ejercicio 3 (3 puntos) Dos números naturales x e y son **primos entre sí** si el único divisor común es el 1.

- Escribe una función booleana que determine si dos números dados son primos entre sí.
- Escribe un procedimiento que pida dos números por la pantalla y los lea del teclado.
- Escribe una función `main()` que lea pares de números positivos con el procedimiento definido en b) y determine si son primos entre sí con la función definida en a) hasta que se introduzcan dos ceros por el teclado. Cuando se tecleen dos ceros, el programa debe terminar **sin llamar** a la función definida en a).

Ejercicio 4 (2 puntos) Escribe una función que calcule la integral definida de la función $f(x) = \cos^2(x)$ en un intervalo $[a, b]$. Debe recibir un parámetro implícito que indique el intervalo mínimo a considerar (por defecto 10^{-4}).

Ejercicio 5 (1 punto) Escribe qué resultado producirá en la pantalla el siguiente programa:

```
#include <iostream>                                int main() {  
using namespace std;                            int w = 1;  
int incr(int& p, int q) {                       for (w=10; w>0; w--) {  
    static int r=0;                             cout << w << " "; cout << incr(w,w) << endl;  
    r += (q+p); p--;  
    return r;  
}  
}  
}
```