

Exercício 1

- O nome do projeto deve ser “Ex1_(Número aluno)”.
- O projeto com o exercício deve ser entregue no Fénix numa pasta com o formato *.zip*.
- A função `abs(·)` disponível no pacote *cmath* é a função módulo e podem usá-la se necessário.
- A pergunta 2 deve ser respondida **no enunciado**.

Nome: _____ Número: _____

1. Considere a série abaixo onde x é um qualquer valor real

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

Escreva um programa que:

- (a) (2 valores) Leia um número inteiro positivo n e faça a sua validação. Isto é, caso o número introduzido não seja positivo, peça sucessivamente valores ao utilizador até obter um número positivo.
- (b) (2 valores) O valor de x deve pertencer ao intervalo $] - 1, 1[$. Leia o valor de x fazendo a sua validação. Isto é, o programa deve pedir sucessivamente valores ao utilizador até ser introduzido um valor válido.
- (c) (3 valores) Calcule o valor da soma parcial da série considerando os valores de n e x introduzidos pelo utilizador nas alíneas anteriores, isto é, calcule

$$S_n = \sum_{i=0}^n (-1)^i x^{2i}.$$

- (d) (1,5 valores) Calcule a soma da série para o valor de x introduzido pelo utilizador, isto é,

$$S = \sum_{i=0}^{\infty} (-1)^i x^{2i} = \frac{1}{1 + x^2}.$$

- (e) (4 valores) Considerando o valor de x introduzido, determine o menor número de parcelas (seja ele k) que é necessário somar de tal forma que

$$|S_k - S| = \left| \sum_{i=0}^k (-1)^i x^{2i} - \frac{1}{1 + x^2} \right| < 0.0001.$$

- (f) (4,5 valores) Calcule a soma dos números primos existentes no intervalo $[1, k]$ e indique quantos números primos aí existem. **Nota:** Caso não tenha resolvido a alínea anterior considere $k = 7$.

As restante cotação é atribuída à apresentação do output indicado em baixo (0,5 valores) e ao facto do programa não conter erros de compilação (0,5 valores).

Espera-se que para os valores $n = 5$ e $x = 0.5$ o programa gere o seguinte output:

n: 5

x: 0.5

Sn = 0.799805

S = 0.8

Numero de parcelas (k) = 7

Existem 4 numeros primos entre 1 e 7 e a sua soma e 17

2. (2 valores) O seguinte excerto de código foi desenvolvido para calcular a soma dos 10 primeiros números ímpares positivos, contudo contém 2 erros. Identifique-os e corrija-os.

```
int nImpares = 0;
int i = 0;
int soma = 0;
while(nImpares <= 10){
    if(i % 2 != 0){
        ++nImpares;
        ++i;
        soma += i;
    }
}
```

