

Aula prática #6 – Funções / Apontadores

Problema 1

O seno de x pode ser calculado usando o desenvolvimento em série de Taylor:

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots + (-1)^{n+1} \frac{x^{2n-1}}{(2n-1)!}, n = 1, 2, 3, \dots$$

Escreva e teste uma função que determine o seno de um ângulo x (introduzido pelo utilizador em radianos), usando este desenvolvimento em série, até um termo em que a diferença para o termo seguinte tenha um valor absoluto abaixo de uma tolerância (especificada pelo utilizador). Sugestão: utilize uma função para calcular o fatorial de um número, semelhante à apresentada na aula teórica 9.

Exemplo

```
1 Qual o valor de x? 1.57
2 Qual o valor da tolerancia? 0.00005
3 O seno de 1.57 e' 1.000003
```

Nota: Repare que $1.57 = \pi/2$.

Problema 2

Escreva uma função que conte quantas vezes um determinado dígito existe num número pedido ao utilizador.

Exemplo

```
1 Introduza um numero inteiro: 3276022
2 introduza um digito: 2
3 O digito 2 aparece 3 vezes no numero 3276022
```

Problema 3

Considere o programa em baixo. Pretende-se que o programa imprima os valores de **var1** e **var2** e os seus endereços (guardados em **ptr1** e **ptr2**).

```

1  #include <stdio.h>
2  int main()
3  {
4      int var1 = 5;
5      char var2 = 'a';
6      int *ptr1 = &var1;
7      char *ptr2;
8      *ptr2 = 'b';
9      printf("var1 tem o endereço %p e o valor %d\n", ptr1, *ptr1);
10     printf("var2 tem o endereço %p e o valor %d\n", ptr2, *ptr2);
11 }

```

Siga os seguintes passos e interprete para cada caso os valores apresentados pelo programa:

- Se tentar compilar e executar este programa ocorrerá o erro "Segmentation fault". Corrija o programa para que isso não aconteça.
- Altere o programa para que este imprima também o tamanho dos tipos de variáveis char, char*, int e int*.
- Altere o programa para que este imprima ainda os valores de **ptr1+1** e **ptr2+1** e compare com os valores de **ptr1** e **ptr2**.

Problema 4

Dado o seguinte programa complete as tabelas:

```

1  #include <stdio.h>
2  int main()
3  {
4      int i, j, *p_1, *p_2, **p_p_1, **p_p_2;
5      i = 4;
6      j = 5;
7      p_1 = &i;
8      p_2 = &j;
9      p_p_1 = &p_2;
10     p_p_2 = &p_1;
11 }

```

Variável	i	j	p_1	p_2	p_p_1	p_p_2
Conteúdo	4	5				
Endereço	1000	1007	1030	1053	1071	1079

Expressão	i	*p_2	&i	&p_2	**p_p_1	*p_p_2	&(*p_1)	j	*p_1	*(&p_1)
Conteúdo										