

# Teste, 20 de Maio de 2023

1. Implemente a função `int perfeito(int x)` que testa se um número inteiro é perfeito, isto é, se é igual à soma dos seus divisores próprios. Por exemplo, 28 é um número perfeito, uma vez que os seus divisores próprios são 1, 2, 4, 7 e 14 ( $1+2+4+7+14=28$ ).
2. Implemente a função `void ordena(Ponto pos[], int N)` que dado um array com N pontos ordena esses pontos por ordem crescente da distância à origem. Por exemplo se o array for  `{{3,3},{2,1},{-1,0}}` depois de ordenado deverá ficar com o conteúdo  `{{-1,0},{2,1},{3,3}}`. O tipo Ponto é definido da seguinte forma (note que as coordenadas dos pontos são números inteiros).

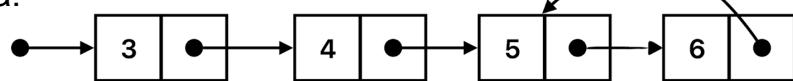
```
typedef struct {
    int x,y;
} Ponto;
```

3. Apresente uma definição da função `int depth(ABin a, int x)` que devolve o menor nível a que um elemento x se encontra na árvore (ou -1 se x não se encontra na árvore). Considere a definição usual do tipo ABin. Considere ainda que a raiz se encontra no nível 0.

```
typedef struct abin_nodo {
    int valor;
    struct abin_nodo *esq, *dir;
} *ABin;
```

4. Implemente a função `int wordle(char secreta[], char tentativa[])` que dada uma palavra secreta que se pretende descobrir e uma tentativa com o mesmo tamanho devolve o número de caracteres na palavra tentativa em que o utilizador já acertou. Ambas as palavras só contêm letras minúsculas. A função deve também modificar a tentativa substituindo todas as letras que não tem correspondente na palavra secreta por um '\*' e convertendo para maiúscula as letras que estão na posição certa. Por exemplo se a palavra secreta for "laranja" e a tentativa for "cerejas" a função deve devolver 1 e alterar a tentativa para "\*\*R\*j\*a\*" (apenas o 'r' está na posição certa e os caracteres 'j' e 'a' aparecem no segredo noutras posições). Se a tentativa for "bananas" a função deve devolver 3 e alterar a tentativa para "\*A\*ANa\*".
5. Implemente a função `LInt periodica(char s[])` que dada uma string com uma sequência infinita periódica de dígitos constrói uma lista (circular) com esses dígitos.

Assuma que a parte da sequência que se repete indefinidamente está representada entre parênteses e aparece sempre no final da string. Assuma também a definição usual do tipo LInt. Por exemplo, se a string for "34(56)" deverá ser construída a seguinte lista.



```
typedef struct lint_nodo {  
    int valor;  
    struct lint_nodo *prox;  
} *LInt;
```