

## **Lista de Exercícios 2 - Ponteiros e Alocação Dinâmica de Memória**

**Bacharelado em Ciência da Computação**

**Linguagens de Programação II**

**Professora:** Susana Marrero Iglesias

1. Escreva um programa que leia um valor de ponto flutuante via teclado. Crie um ponteiro que aponte para o valor. Imprima o valor na tela usando a técnica de referência direta e a técnica de referência indireta. Imprima também o endereço da variável apontada.
2. Crie 3 variáveis e associe um valor a cada uma delas. Crie 3 ponteiros, um para cada variável, e mostre o conteúdo das variáveis através destes ponteiros.
3. Crie 3 variáveis e associe um valor a cada uma delas. Crie 1 ponteiro e mostre o conteúdo das 3 variáveis através deste único ponteiro.
4. Escreva um programa incluindo as declarações seguintes.

```
char key, choice;  
int num, count;  
long int date;  
float yield;  
double price;
```

- a) Use no programa o operador de endereço e a função `printf()` para exibir os endereços correspondentes a cada variável.
  - b) Desenhe um diagrama de como seu compilador reservou o armazenamento pelas variáveis no programa. Em seu diagrama, preencha os endereços exibidos pelo programa.
  - c) Modifique o programa escrito para (a) exibindo a quantidade de memória que seu computador reserva para cada tipo de dado (use o operador `sizeof`). Com esta informação e a informação do endereço provida em (b), determine se seu compilador reservou o armazenamento pelas variáveis na ordem pelo qual elas foram declaradas. Comente os resultados.
5. Uma turma tem 30 alunos. Faça um programa que receba as notas dos alunos relativas as 2 provas que eles realizaram durante o semestre. Exiba a média de cada aluno, a situação (se média maior ou igual a 7, aluno aprovado), e a média da turma para cada prova. Utilize 2 vetores alocados dinamicamente para armazenar as notas, referentes a cada prova.
  6. Altere o programa acima para que seja possível processar as notas de turmas com qualquer número de alunos.

7. Escreva um programa que declara três únicos vetores dimensionais chamados `Kms`, `Lts` e `KpL`. Cada vetor deve ter capacidade para armazenar 10 elementos. No vetor `Kms` armazene os números: 240.5, 300.0, 189.6, 310.6, 280.7, 216.9, 199.4, 160.3, 177.4, 192.3. No vetor `Lts` armazene os números: 10.3, 15.6, 8.7, 14, 16.3, 15.7, 14.9, 10.7, 8.3, 8.4. Cada elemento do vetor `KpL` deve ser calculado com os correspondentes elementos dos vetores `Kms` e `Lts`; por exemplo,  $KpL[0] = Kms[0]/Lts[0]$ . Calcule os elementos do vetor `KpL` e mostre eles na tela.
  - a) Escreva o programa usando a notação convencional de subscritos.
  - b) Escreva o programa usando aritmética de ponteiros.
8. Escreve um programa que leia  $n$  elementos de um vetor de inteiros e informe:
  - a) a média dos elementos do vetor,
  - b) o maior elemento do vetor,
  - c) o menor elemento do vetor,

a quantidade de elementos  $n$  é informada pelo usuário. Utilize alocação dinâmica para reservar memória para os elementos do vetor. Para informar o maior e menor elemento utilize ponteiros que apontem aos respectivos elementos. Utilize a notação de subscritos para acessar os elementos do vetor.
9. Modifique o programa do exercício anterior de tal forma que o acesso aos elementos do vetor seja feito utilizando aritmética de ponteiros.
10. Escreva um programa que receba como dados duas matrizes  $A$  e  $B$ , de  $m$  linhas e  $n$  colunas cada uma. As dimensões das matrizes são informadas pelo usuário em tempo de execução. Imprima na tela o resultado de  $A + B$  e  $A - B$ .
11. Escreva um programa que leia uma matriz quadrada via teclado e informe:
  - a) os dois maiores elementos da matriz,
  - b) a quantidade de elementos negativos,
  - c) o produto de todos os elementos da matriz.

Seu programa deve funcionar para matrizes de quaisquer dimensão.
12. Escreva um programa que receba como entrada duas matrizes e imprima na tela o produto delas. Seu programa deve funcionar para matrizes de quaisquer dimensão.
13. Escreva um programa que receba uma cadeia de caracteres (`str`) e informe a quantidade de caracteres. Utilize um ponteiro a caractere para processar os elementos de `str`.
14. Escreva um programa que receba uma cadeia de caracteres `str` e um caractere `c`. Seu programa deve informar a quantidade de ocorrências de `c` em `str`. Utilize aritmética de ponteiros para acessar os elementos de `str`.

15. Escreva um programa que receba uma cadeia de caracteres e informe:

- a) a quantidade de vogais,
- b) a quantidade de dígitos,
- c) e a quantidade de caracteres em branco (espaços, tabulação e nova linha).

Utilize aritmética de ponteiros para acessar os elementos da cadeia de caracteres.

16. Modifique o programa anterior (15a) de forma que a quantidades de vogais seja mostrada graficamente (histograma). Para a frase “*Adoro ponteiros em linguagem C...*” deve mostrar:

```
a **
e ***
i **
o ****
u *
```

17. Escreva um programa que receba um vetor de  $n$  números inteiros ( $n$  é digitado pelo usuário). Seu programa deve informar os valores que se repetem e a quantidade de vezes que cada um se repete. Exemplo: para o vetor

1, 2, 4, 1, 4, 5, 4

seu programa informa

```
1 se repete 2 vezes
4 se repete tres vezes .
```

18. Modifique o programa anterior para informar os números que não se repetem.