

Linguagem de Programação

Lista Avaliativa 2

Análise e Desenvolvimento de Sistemas – 2º Ciclo Noturno

Gabriel Rodrigues de Oliveira

- 1) Escreva um programa que leia a idade de 10 pessoas e armazene-as em um vetor. Calcule e mostre:
 - a) a menor idade
 - b) a média das idades
 - c) a quantidade de pessoas que tem idade entre 20 e 30 anos (inclusive)
 - d) a quantidade de pessoas com idade maior que a média

```
1 #include <stdio.h>
2 #include <locale.h>
3
4 int main(){
5     setlocale(LC_ALL, "Portuguese");
6     int idade[10], idade_menor, qtd_20e30 = 0, soma_20e30 = 0, acima_media = 0;
7     float media, soma = 0;
8
9     // definir os valores dentro do vetor -> idade
10    for(int i = 0; i < 10; i++){
11        printf("Digite sua idade: ");
12        scanf("%i", &idade[i]);
13    }
14    idade_menor = idade[0];
15    for(int i = 0; i < 10; i++){
16        soma += idade[i]; // soma de idades para definir a média
17        // descobrir a menor idade digitada
18        if(idade[i] < idade_menor){
19            idade_menor = idade[i];
20        }
21        // idades entre 20 e 30 anos
22        if(idade[i] >= 20 && idade[i] <= 30){
23            qtd_20e30++;
24            soma_20e30 += idade[i];
25        }
26    }
27    media = soma / 10; // encontrar a média
28    // definir idades maiores que a média
29    for(int i = 0; i < 10; i++){
30        if(idade[i] > media){
31            acima_media++;
32        }
33    }
34    printf("\nMenor idade: %i\n", idade_menor);
35    printf("Média das idades: %.1f\n", media);
36    printf("Quantidade de pessoas entre 20 e 30 anos: %i\n", qtd_20e30);
37    printf("Quantidade de pessoas com idades maiores que a média: %i", acima_media);
38    return 0;
39 }
```

C:\Users\gabri\OneDrive\Doc x + v

Digite sua idade: 18
Digite sua idade: 20
Digite sua idade: 29
Digite sua idade: 35
Digite sua idade: 36
Digite sua idade: 40
Digite sua idade: 44
Digite sua idade: 48
Digite sua idade: 50
Digite sua idade: 62

Menor idade: 18
Média das idades: 13,8
Quantidade de pessoas entre 20 e 30 anos: 2
Quantidade de pessoas com idades maiores que a média: 10

Process exited after 27.73 seconds with return value 0
Pressione qualquer tecla para continuar. . . |

2) Faça um programa que preencha um vetor com 10 cores diferentes. Depois permita fazer uma pesquisa se uma determinada cor existe armazenada no vetor, se existir deve ser impresso na tela a cor e em qual posição (índice) esta cor está armazenada. A pesquisa deve ser feita até que seja digitado FIM na cor a ser pesquisada na lista.

```
1 #include <locale.h>
2 #include <iostream>
3 #include <string>
4
5 int main() {
6     setlocale(LC_ALL, "Portuguese");
7     // Declara o vetor, a variável de pesquisa e a posição que a cor se encontra (caso encontrada)
8     std::string cores[10] = {"vermelho", "laranja", "amarelo", "verde", "azul", "roxo", "rosa", "marrom", "cinza", "preto"};
9     std::string pesquisa;
10    int pos;
11
12    while(true){
13        // Solicita ao usuário a cor a ser buscada
14        std::cout << "\nDigite a cor que deseja pesquisar ou digite 'FIM' para encerrar o programa: ";
15        std::cin >> pesquisa;
16        // Verifica se o usuário digitou "Fim para encerrar o programa"
17        if(pesquisa == "FIM"){
18            std::cout << "Fim do Programa..." << std::endl;
19            break;
20        }
21        // Define que a cor não foi encontrada
22        bool encontrada = false;
23        // Pesquisa a cor
24        for(int i = 0; i < 10; i++){
25            if (cores[i] == pesquisa){
26                encontrada = true;
27                pos = i+1;
28                break;
29            }
30        }
31        // Imprime a cor encontrada | Informa que a cor não foi encontrada
32        if(encontrada){
33            std::cout << "A cor " << pesquisa << " foi encontrada na " << pos << "ª posição do vetor de cores." << std::endl;
34        }
35        else{
36            std::cout << "A cor " << pesquisa << " não foi encontrada no vetor de cores." << std::endl;
37        }
38    }
39    return 0;
40 }
```

C:\Users\gabri\OneDrive\Doc x + v - □ x

Digite a cor que deseja pesquisar ou digite 'FIM' para encerrar o programa: preto
A cor preto foi encontrada na 10ª posição do vetor de cores.

Digite a cor que deseja pesquisar ou digite 'FIM' para encerrar o programa: dourado
A cor dourado não foi encontrada no vetor de cores.

Digite a cor que deseja pesquisar ou digite 'FIM' para encerrar o programa: azul
A cor azul foi encontrada na 5ª posição do vetor de cores.

Digite a cor que deseja pesquisar ou digite 'FIM' para encerrar o programa: verde
A cor verde foi encontrada na 4ª posição do vetor de cores.

Digite a cor que deseja pesquisar ou digite 'FIM' para encerrar o programa: FIM
Fim do Programa...

Process exited after 24.78 seconds with return value 0
Pressione qualquer tecla para continuar. . . |

3) Faça um programa que preencha dois vetores, A e B com 5 números em cada. Gere o vetor C, com os números do vetor A e B. Depois calcule e mostre na tela a quantidade de números perfeitos. Um número é perfeito quando ele é igual à soma dos seus divisores exceto ele próprio. (Ex: 6 é perfeito, $6 = 1 + 2 + 3$, que são seus divisores).

```
1 #include <stdio.h>
2 #include <locale.h>
3
4 int main(){
5     setlocale(LC_ALL, "Portuguese");
6     int vetA[5], vetB[5], vetC[10], div, perfeito = 0;
7     // armazena os valores do 1º vetor -> vetA
8     for(int i = 0; i < 5; i++){
9         printf("Digite os números do 1º vetor: ");
10        scanf("%i", &vetA[i]);
11    }
12    printf("\n-----\n");
13    // armazena os valores do 2º vetor -> vetB
14    for(int i = 0; i < 5; i++){
15        printf("Digite os números do 2º vetor: ");
16        scanf("%i", &vetB[i]);
17    }
18    // entrelaça os valores do 1º e 2º vetor para o 3º vetor -> vetC
19    int k = 0;
20    for(int i = 0; i < 5; i++){
21        vetC[k] = vetA[i];
22        k++;
23        vetC[k] = vetB[i];
24        k++;
25    }
26    // encontra os números perfeitos
27    for(int i = 0; i < vetC[i]; i++){
28        // encontrar os divisores
29        div = 0;
30        for(int k = 1; k < vetC[i]; k++){
31            if(vetC[i] % k == 0){
32                div += k;
33            }
34        }
35        // comparar a soma dos divisores com o número do vetor
36        if(vetC[i] == div){
37            perfeito++;
38        }
39    }
40    printf("\nA quantidade de números perfeitos dentro do vetor é: %i", perfeito);
41    return 0;
42 }
```

```
C:\Users\gabri\OneDrive\Doc  X  +  v
Digite os números do 1º vetor: 5
Digite os números do 1º vetor: 6
Digite os números do 1º vetor: 3
Digite os números do 1º vetor: 5
Digite os números do 1º vetor: 4
```

```
-----
Digite os números do 2º vetor: 6
Digite os números do 2º vetor: 8
Digite os números do 2º vetor: 9
Digite os números do 2º vetor: 2
Digite os números do 2º vetor: 1
```

```
A quantidade de números perfeitos dentro do vetor é: 2
```

```
-----
Process exited after 7.422 seconds with return value 0
Pressione qualquer tecla para continuar. . . |
```

- 4) Faça um programa que preencha um array com os nomes de 5 produtos, e outro array com o valor dos produtos. Calcule e mostre:
- a quantidade de produtos que o valor é abaixo de 10 reais;
 - a média dos valores dos produtos;
 - a quantidade de produtos que valor acima da média;
 - a maior valor e o nome do produto;
 - faça uma listagem que imprima na tela (Nome Vlr do produto)

```
1 #include <locale.h>
2 #include <iostream>
3 #include <string>
4
5 int main(){
6     setlocale(LC_ALL, "Portuguese");
7     std::string nomes[5], maiorProd;
8     float valores[5], media, maiorValor;
9     int abaixoDe10 = 0, acimaMedia = 0;
10    for(int i = 0; i < 5; i++){
11        std::cout << "Nome do produto: "; std::cin >> nomes[i];
12        std::cout << "Valor: R$"; std::cin >> valores[i];
13    }
14    maiorValor = valores[0];
15    maiorProd = nomes[0];
16    for(int i = 0; i < 5; i++){
17        // Encontra os valores menores que 10
18        if(valores[i] < 10){
19            abaixoDe10++;
20        }
21        media += valores[i]; // Soma da média
22        // Compara os valores para descobrir o maior valor e o nome
23        if(valores[i] > maiorValor){
24            maiorValor = valores[i];
25            maiorProd = nomes[i];
26        }
27    }
28    // Valores acima da média
29    media = media / 5;
30    for(int i = 0; i < 5; i++){
31        if(valores[i] > media){
32            acimaMedia++;
33        }
34    }
35    // Listagem com nomes e valores dos produtos
36    std::cout << "-----\n" << "Nomes          Vlr do Produto" << std::endl;
37    for(int i = 0; i < 5; i++){
38        std::cout << nomes[i] << "          R$" << valores[i] << std::endl;
39    }
40    std::cout << "\n-----\nQuant de prod abaixo de R$10,00: " << abaixoDe10 << std::endl;
41    std::cout << "Média dos valores: R$" << media << std::endl;
42    std::cout << "Quant de prod acima da média: " << acimaMedia << std::endl;
43    std::cout << "Prod com maior valor: " << maiorProd << ", R$" << maiorValor << std::endl;
44    return 0;
45 }
```

```
C:\Users\gabri\OneDrive\Doc  X + v
Nome do produto: Batata
Valor: R$9.80
Nome do produto: Uva
Valor: R$6.99
Nome do produto: Suco
Valor: R$4.29
Nome do produto: Arroz
Valor: R$24.80
Nome do produto: Carne
Valor: R$40.99
-----
Nomes          Vlr do Produto
Batata         R$9.8
Uva            R$6.99
Suco           R$4.29
Arroz          R$24.8
Carne          R$40.99
-----
```

```
Quant de prod abaixo de R$10,00: 3
Média dos valores: R$17.374
Quant de prod acima da média: 2
Prod com maior valor: Carne, R$40.99
-----
```

```
Process exited after 65.86 seconds with return value 0
Pressione qualquer tecla para continuar. . . |
```

5) Escreva um programa que preencha um vetor com os nomes de 5 vendedores, preencha também outro vetor com valor total das vendas de cada vendedor. Cada vendedor recebe 10% de comissão sobre as vendas. Faça os seguintes cálculos e mostre os resultados na tela:

- Uma listagem com o nome e o valor a receber de cada vendedor (total das vendas * 0.10)
- O total (bruto) vendido pelos 5 vendedores
- A média do total de vendas (valor bruto vendido por cada vendedor)
- A quantidade de vendedores que venderam acima da média das vendas
- O maior valor de comissão e o nome do vendedor que recebeu

```

1  #include <locale.h>
2  #include <iostream>
3  #include <string>
4
5  int main(){
6      setlocale(LC_ALL, "Portuguese");
7      std::string nomes[5], maiorVend;
8      float totalVendas[5], comissao[5], media = 0, total = 0, maiorComi;
9      int acimaMedia = 0;
10     // Armazena os valores nos vetores, media e total
11     for(int i = 0; i < 5; i++){
12         std::cout << "Nome do Vendedor: "; std::cin >> nomes[i];
13         std::cout << "Total de Vendas: R$"; std::cin >> totalVendas[i];
14         comissao[i] = totalVendas[i] * 0.10;
15         media += totalVendas[i];
16         total += totalVendas[i];
17     }
18     // Media de vendas
19     media /= 5;
20     // Listagem: Nomes | Valor a Receber
21     std::cout << "\n-----\nNomes          Valor a Receber" << std::endl;
22     for(int i = 0; i < 5; i++){
23         std::cout << nomes[i] << "          R$" << comissao[i] << std::endl;
24         // Quantidade de vendas acima da média
25         if(totalVendas[i] > media){
26             acimaMedia++;
27         }
28         // Maior comissão e vendedor
29         if(i == 0 || comissao[i] > maiorComi){
30             maiorComi = comissao[i];
31             maiorVend = nomes[i];
32         }
33     }
34     std::cout << "\n-----\nTotal Bruto: R$" << total << std::endl;
35     std::cout << "Média total de vendas: R$" << media << std::endl;
36     std::cout << "Quantidade de vendedores que venderam acima da média: " << acimaMedia << std::endl;
37     std::cout << "Maior valor de comissão e vendedor: " << maiorComi << ", " << maiorVend << std::endl;
38     return 0;
39 }

```

C:\Users\gabriel\OneDrive\Doc

Nome do Vendedor: Ryan
Total de Vendas: R\$1000
Nome do Vendedor: Luis
Total de Vendas: R\$2000
Nome do Vendedor: Leticia
Total de Vendas: R\$3000
Nome do Vendedor: Gabriel
Total de Vendas: R\$2500
Nome do Vendedor: Gustavo
Total de Vendas: R\$1500

Nomes	Valor a Receber
Ryan	R\$100
Luis	R\$200
Leticia	R\$300
Gabriel	R\$250
Gustavo	R\$150

Total Bruto: R\$10000
Média total de vendas: R\$2000
Quantidade de vendedores que venderam acima da média: 2
Maior valor de comissão e vendedor: 300, Leticia

Process exited after 32.91 seconds with return value 0
Pressione qualquer tecla para continuar. . . |

- 6) Escreva um programa que leia o nome e o peso de 10 pessoas e armazene-as em vetores. Calcule e mostre:
- o menor peso e o nome da pessoa
 - a média dos pesos
 - Mostre na tela um relatório com o nome e peso das pessoas cujo peso é inferior à média

```

1  #include <locale.h>
2  #include <iostream>
3  #include <string>
4
5  int main(){
6      setlocale(LC_ALL, "Portuguese");
7      std::string nome[10];
8      std::string nome_menorP;
9      int peso[10], menorP, media = 0, soma = 0;
10     // Armazenar os nomes e os pesos
11     for(int i = 0; i < 10; i++){
12         std::cout << "Digite o " << i+1 << "º nome: ";
13         std::cin >> nome[i];
14         std::cout << "Digite o " << i+1 << "º peso: ";
15         std::cin >> peso[i];
16     }
17     menorP = peso[0];
18     nome_menorP = nome[0];
19     for(int i = 0; i < 10; i++){
20         // menor peso e o nome
21         if(peso[i] < menorP){
22             menorP = peso[i];
23             nome_menorP = nome[i];
24         }
25         // valores para tirar a média
26         soma += peso[i];
27     }
28     // printar o menor peso e a quem pertence
29     std::cout << "O menor peso pertence à " << nome_menorP << ", com " << menorP << "kg." << std::endl;
30     // definir e printar a média de pesos
31     media = soma / 10;
32     std::cout << "A média de pesos é: " << media << "kg." << std::endl;
33     // printar as pessoas com o peso menor que a média
34     for(int i = 0; i < 10; i++){
35         if(peso[i] < media){
36             std::cout << nome[i] << " com " << peso[i] << "kg, tem o peso menor que a média de " << media << "kg." << std::endl;
37         }
38     }
39     return 0;
40 }

```

```

Digite o 1º nome: Gabriel
Digite o 1º peso: 55
Digite o 2º nome: Lucas
Digite o 2º peso: 60
Digite o 3º nome: Daniel
Digite o 3º peso: 77
Digite o 4º nome: Leticia
Digite o 4º peso: 60
Digite o 5º nome: Leonardo
Digite o 5º peso: 80
Digite o 6º nome: Rebeca
Digite o 6º peso: 65
Digite o 7º nome: Gustavo
Digite o 7º peso: 73
Digite o 8º nome: Denis
Digite o 8º peso: 90
Digite o 9º nome: Luis
Digite o 9º peso: 160
Digite o 10º nome: Victor
Digite o 10º peso: 45
O menor peso pertence à Victor, com 45kg.
A média de pesos é: 76kg.
Gabriel com 55kg, tem o peso menor que a média de 76kg.
Lucas com 60kg, tem o peso menor que a média de 76kg.
Leticia com 60kg, tem o peso menor que a média de 76kg.
Rebeca com 65kg, tem o peso menor que a média de 76kg.
Gustavo com 73kg, tem o peso menor que a média de 76kg.
Victor com 45kg, tem o peso menor que a média de 76kg.

```

```

-----
Process exited after 92.17 seconds with return value 0
Pressione qualquer tecla para continuar. . . |

```

7) Faça um programa que receba dez números inteiros e armazene em um vetor. Calcule e mostre:

- A soma dos números primos
- A média dos números múltiplos de 3
- A quantidade de números ímpares e maiores que 10

```
1 #include <stdio.h>
2 #include <locale.h>
3
4 int main(){
5     setlocale(LC_ALL, "Portuguese");
6     int num[10], div, qtd_impar = 0;
7     float soma_primos = 0, media, soma = 0, qtd = 0;
8     for(int i = 0; i < 10; i++){
9         printf("Digite o %iº número: ", i+1);
10        scanf("%i", &num[i]);
11    }
12    for(int i = 0; i < 10; i++){
13        // Encontrar os ímpares e maiores que 10
14        if(num[i] % 2 == 0){ // Verificar se é ímpar ou par
15            qtd_impar = qtd_impar;
16        } else if(num[i] > 10){ // Definir os ímpares maiores que 10
17            qtd_impar++;
18        }
19        // Encontrar os múltiplos de 3
20        if(num[i] % 3 == 0){
21            qtd++;
22            soma += num[i];
23        }
24        // Encontrar os primos
25        div = 0;
26        for(int k = 1; k <= num[i]; k++){
27            if(num[i] % k == 0){
28                div++;
29            }
30        }
31        if(div == 2){
32            soma_primos += num[i];
33        }
34    }
35    // Média dos múltiplos de 3
36    if(qtd > 0){
37        media = soma / qtd;
38    } else{
39        media = 0;
40    }
41    printf("\nSoma dos primos: %.0f", soma_primos);
42    printf("\nMédia dos números múltiplos de 3: %.1f", media);
43    printf("\nQuantidade dos ímpares maiores que 10: %i", qtd_impar);
44    return 0;
45 }
```

C:\Users\gabri\OneDrive\Doc x + v

Digite o 1º número: 2
Digite o 2º número: 5
Digite o 3º número: 6
Digite o 4º número: 11
Digite o 5º número: 25
Digite o 6º número: 30
Digite o 7º número: 22
Digite o 8º número: 16
Digite o 9º número: 28
Digite o 10º número: 42

Soma dos primos: 18
Média dos números múltiplos de 3: 26,0
Quantidade dos ímpares maiores que 10: 2

Process exited after 24.18 seconds with return value 0
Pressione qualquer tecla para continuar. . . |