

Gabarito – Livro: Programação de Computadores com C++

Prezado(a) Educador(a)

O material ora apresentado caracteriza-se em ser formado pelas respostas dos exercícios de fixação do livro em referência.

As respostas aqui dadas são fornecidas genericamente no sentido de atender a solução de certo problema do ponto de vista algorítmico e servirá como ponto de auxílio ao seu trabalho.

As respostas aqui indicadas são respostas propostas sem a pretensão de serem as únicas ou as melhores respostas.

As respostas dadas são respostas possíveis.

É pertinente salientar que esta obra deve ser utilizada em conjunto ou após o estudo do livro Algoritmos: Lógica para o Desenvolvimento de Programação de Computadores, pois alguns pontos não comentados nesta obra são encontrados na obra sobre algoritmos.

O conjunto de exercícios aqui apresentados é em média suficiente para a fixação dos detalhes sobre o uso da linguagem de programação de computadores em foco na mente do educando.

Espero com isso estar fornecendo uma maior facilidade para o(a) colega poder ministrar a base de conhecimento sobre programação de computadores.

Atenciosamente

José Augusto N. G. Manzano

Capítulo 2

Exercício A

```
#include <iostream>
#include <iomanip>

using namespace std;

int main(void)
{
    float TEMPO, VELOCIDADE, DISTANCIA, LITROS_USADOS;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> TEMPO;
    cin >> VELOCIDADE;
    DISTANCIA = TEMPO * VELOCIDADE;
    LITROS_USADOS = DISTANCIA / 12;
    cout << setw(8) << TEMPO << endl;
    cout << setw(8) << VELOCIDADE << endl;
    cout << setw(8) << DISTANCIA << endl;
    cout << setw(8) << LITROS_USADOS << endl;
    return 0;
}
```

Exercício B

```
#include <iostream>
#include <iomanip>

using namespace std;

int main(void)
{
    float C, F;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> C;
    F = (9 * C + 160) / 5;
    cout << setw(8) << F << endl;
    return 0;
}
```

Exercício C

```
#include <iostream>
#include <iomanip>

using namespace std;

int main(void)
{
    float F, C;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> F;
    C = ((F - 32) * 5) / 9;
    cout << setw(8) << C << endl;
}
```

```
    return 0;
}
```

Exercício D

```
#include <iostream>
#include <iomanip>

using namespace std;

int main(void)
{
    float C, K;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> C;
    K = C + 273.15;
    cout << setw(8) << K << endl;
    return 0;
}
```

Exercício E

```
#include <iostream>
#include <iomanip>

using namespace std;

int main(void)
{
    float K, C;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> K;
    C = K - 273.15;
    cout << setw(8) << C << endl;
    return 0;
}
```

Exercício F

```
#include <iostream>
#include <iomanip>

using namespace std;

int main(void)
{
    float F, K;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> F;
    K = ((F - 32) * 5) / 9 + 273.15;
    cout << setw(8) << K << endl;
    return 0;
}
```

Exercício G

```
#include <iostream>
#include <iomanip>
```

```
using namespace std;

int main(void)
{
    float K, F;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> K;
    F = (9 * (K - 273.15) + 160) / 5;
    cout << setw(8) << F << endl;
    return 0;
}
```

Exercício H

```
#include <iostream>
#include <iomanip>

using namespace std;

int main(void)
{
    float V, R, A;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> R;
    cin >> A;
    V = 3.141592653589793 * R * R * A;
    cout << setw(8) << V << endl;
    return 0;
}
```

Exercício I

```
#include <iostream>
#include <iomanip>

using namespace std;

int main(void)
{
    float P, D, R;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> D;
    cin >> R;
    P = D * 4 * 3.141592653589793 * R * R * R / 3;
    cout << setw(8) << P << endl;
    return 0;
}
```

Exercício J

```
#include <iostream>

using namespace std;

int main(void)
{
    int A, B, X;
    cin >> A;
    cin >> B;
    X = A;
```

```
A = B;
B = X;
cout << A << endl;
cout << B << endl;
return 0;
}
```

Capítulo 3

Exercício A

```
#include <iostream>
using namespace std;

int main(void)
{
    int A, B, DIF;
    cin >> A;
    cin >> B;
    if (A > B)
        DIF = A - B;
    else
        DIF = B - A;
    cout << DIF << endl;
    return 0;
}
```

Exercício B

```
#include <iostream>
using namespace std;

int main(void)
{
    int N, NP;
    cin >> N;
    if (N < 0)
        NP = N * -1;
    else
        NP = N;
    cout << NP << endl;
    return 0;
}
```

Exercício C

```
#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    float N1, N2, N3, N4, MD;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> N1;
    cin >> N2;
    cin >> N3;
    cin >> N4;
    MD = (N1 + N2 + N3 + N4) / 4;
    if (MD >= 5)
        cout << "Aprovado, " << MD << endl;
}
```

```

    else
        cout << "Reprovado, " << MD << endl;
    return 0;
}

```

Exercício D

```

#include <iostream>
using namespace std;

int main(void)
{
    int A, B, C, X;
    cin >> A;
    cin >> B;
    cin >> C;
    if (A > B)
    {
        X = A;
        A = B;
        B = X;
    }
    if (A > C)
    {
        X = A;
        A = C;
        C = X;
    }
    if (B > C)
    {
        X = B;
        B = C;
        C = X;
    }
    cout << A << endl;
    cout << B << endl;
    cout << C << endl;
    return 0;
}

```

Exercício E

```

#include <iostream>
using namespace std;

int main(void)
{
    int A, B, C, D, R2, R3;
    cin >> A;
    cin >> B;
    cin >> C;
    cin >> D;

    R2 = A - 2 * (A / 2); // ou R2 = A % 2;
    R3 = A - 3 * (A / 3); // ou R3 = A % 3;
    if (R2 == 0 and R3 == 0)
        cout << A << endl;

    R2 = B - 2 * (B / 2); // ou R2 = A % 2;
    R3 = B - 3 * (B / 3); // ou R3 = A % 3;
    if (R2 == 0 and R3 == 0)
        cout << B << endl;

    R2 = C - 2 * (C / 2); // ou R2 = A % 2;
    R3 = C - 3 * (C / 3); // ou R3 = A % 3;
    if (R2 == 0 and R3 == 0)
        cout << C << endl;
}

```

```

R2 = D - 2 * (D / 2); // ou R2 = A % 2;
R3 = D - 3 * (D / 3); // ou R3 = A % 3;
if (R2 == 0 and R3 == 0)
    cout << D << endl;

return 0;
}

```

Exercício F

```

#include <iostream>
using namespace std;

int main(void)
{
    int A, B, C, D, R2, R3;
    cin >> A;
    cin >> B;
    cin >> C;
    cin >> D;

    R2 = A - 2 * (A / 2); // ou R2 = A % 2;
    R3 = A - 3 * (A / 3); // ou R3 = A % 3;
    if (R2 == 0 or R3 == 0)
        cout << A << endl;

    R2 = B - 2 * (B / 2); // ou R2 = A % 2;
    R3 = B - 3 * (B / 3); // ou R3 = A % 3;
    if (R2 == 0 or R3 == 0)
        cout << B << endl;

    R2 = C - 2 * (C / 2); // ou R2 = A % 2;
    R3 = C - 3 * (C / 3); // ou R3 = A % 3;
    if (R2 == 0 or R3 == 0)
        cout << C << endl;

    R2 = D - 2 * (D / 2); // ou R2 = A % 2;
    R3 = D - 3 * (D / 3); // ou R3 = A % 3;
    if (R2 == 0 or R3 == 0)
        cout << D << endl;

    return 0;
}

```

Exercício G

```

#include <iostream>
using namespace std;

int main(void)
{
    int N;
    cin >> N;
    if (N >= 1 and N <= 9)
        cout << "O valor esta na faixa permitida.";
    else
        cout << "O valor esta fora da faixa permitida.";
    return 0;
}

```

Exercício H

```

#include <iostream>
using namespace std;

```

```

int main(void)
{
    int A, B, C, D, E, X, Y;
    cin >> A;
    cin >> B;
    cin >> C;
    cin >> D;
    cin >> E;
    X = A;
    if (X > B)
        X = B;
    if (X > C)
        X = C;
    if (X > D)
        X = D;
    if (X > E)
        X = E;
    Y = A;
    if (Y < B)
        Y = B;
    if (Y < C)
        Y = C;
    if (Y < D)
        Y = D;
    if (Y < E)
        Y = E;
    cout << "Maior = " << X << endl;
    cout << "Menor = " << Y << endl;
    return 0;
}

```

Exercício I

```

#include <iostream>
using namespace std;

int main(void)
{
    int N, R;
    cin >> N;
    R = N % 2; // ou R = N - 2 * (N / 2);
    if (R == 0)
        cout << "Par" << endl;
    else
        cout << "Impar" << endl;
    return 0;
}

```

Exercício J

```

#include <iostream>
using namespace std;

int main(void)
{
    int N, R;
    cin >> N;
    R = N * 2;
    if (R > 30)
        cout << R << endl;
    return 0;
}

```

Capítulo 4

Exercício 1A - while

```
#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, QUAD;
    I = 15;
    while (I <= 200)
    {
        QUAD = I * I;
        cout << "Quadrado de " << setw(3) << I << " = " << setw(5) << QUAD << endl;
        I++;
    }
    return 0;
}
```

Exercício 2A - do...while

```
#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, QUAD;
    I = 15;
    do
    {
        QUAD = I * I;
        cout << "Quadrado de " << setw(3) << I << " = " << setw(5) << QUAD << endl;
        I++;
    }
    while (I <= 200);
    return 0;
}
```

Exercício 3A - for

```
#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, QUAD;
    for (I = 15; I <= 200; I++)
    {
        QUAD = I * I;
        cout << "Quadrado de " << setw(3) << I << " = " << setw(5) << QUAD << endl;
    }
    return 0;
}
```

Exercício 1B - while

```
#include <iostream>
#include <iomanip>
using namespace std;
```

```

int main(void)
{
    int I, S;
    S = 0;
    I = 1;
    while (I <= 100)
    {
        S += I;
        I++;
    }
    cout << "Soma = " << setw(5) << S << endl;
    return 0;
}

```

Exercício 2B - do...while

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, S;
    S = 0;
    I = 1;
    do
    {
        S += I;
        I++;
    }
    while (I <= 100);
    cout << "Soma = " << setw(5) << S << endl;
    return 0;
}

```

Exercício 3B - for()

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, S;
    S = 0;
    for (I = 1; I <= 100; I++)
        S += I;
    cout << "Soma = " << setw(5) << S << endl;
    return 0;
}

```

Exercício 1C - while

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, R;
    I = 0;
    while (I <= 20)
    {
        R = I % 2;
        if (R != 0)

```

```

        cout << setw(3) << I << endl;
        I++;
    }
    return 0;
}

```

Exercício 2C - do...while

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, R;
    I = 0;
    do
    {
        R = I % 2;
        if (R != 0)
            cout << setw(3) << I << endl;
        I++;
    }
    while (I <= 20);
    return 0;
}

```

Exercício 3C - for()

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, R;
    for (I = 0; I <= 20; I++)
    {
        R = I % 2;
        if (R != 0)
            cout << setw(3) << I << endl;
    }
    return 0;
}

```

Exercício 1D - while

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, R;
    I = 1;
    while (I < 30)
    {
        R = I % 4;
        if (R == 0)
            cout << setw(3) << I << endl;
        I++;
    }
    return 0;
}

```

Exercício 2D - do...while

```
#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, R;
    I = 1;
    do
    {
        R = I % 4;
        if (R == 0)
            cout << setw(3) << I << endl;
        I++;
    }
    while (I < 30);
    return 0;
}
```

Exercício 3D - for()

```
#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, R;
    for (I = 1; I < 30; I++)
    {
        R = I % 4;
        if (R == 0)
            cout << setw(3) << I << endl;
    }
    return 0;
}
```

Exercício 1E -while

```
#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, R, S;
    S = 0;
    I = 1;
    while (I <= 50)
    {
        R = I % 2;
        if (R == 0)
            S += I;
        I++;
    }
    cout << setw(4) << S << endl;
    return 0;
}
```

Exercício 2E - do...while

```
#include <iostream>
#include <iomanip>
```

```

using namespace std;

int main(void)
{
    int I, R, S;
    S = 0;
    I = 1;
    do
    {
        R = I % 2;
        if (R == 0)
            S += I;
        I++;
    }
    while (I <= 50);
    cout << setw(4) << S << endl;
    return 0;
}

```

Exercício 3E - for()

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, R, S;
    S = 0;
    for (I = 1; I <= 50; I++)
    {
        R = I % 2;
        if (R == 0)
            S += I;
    }
    cout << setw(4) << S << endl;
    return 0;
}

```

Exercício 1F - while

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, F, N;
    F = 1;
    I = 1;
    cout << "Entre o valor da fatorial: "; cin >> N;
    while (I <= N)
    {
        F *= I;
        I++;
    }
    cout << "Fatorial = " << setw(5) << F << endl;
    return 0;
}

```

Exercício 2F - do...while

```

#include <iostream>
#include <iomanip>
using namespace std;

```

```

int main(void)
{
    int I, F, N;
    F = 1;
    I = 1;
    cout << "Entre o valor da fatorial: "; cin >> N;
    do
    {
        F *= I;
        I++;
    }
    while (I <= N);
    cout << "Fatorial = " << setw(5) << F << endl;
    return 0;
}

```

Exercício 3F - for()

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int I, F, N;
    F = 1;
    cout << "Entre o valor da fatorial: "; cin >> N;
    for (I = 1; I <= N; I++)
        F *= I;
    cout << "Fatorial = " << setw(5) << F << endl;
    return 0;
}

```

Exercício 1G - while

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int B, E, P, I;
    cout << "Entre o valor da base .....: "; cin >> B;
    cout << "Entre o valor da expoente ...: "; cin >> E;
    P = 1;
    I = 1;
    while (I <= E)
    {
        P *= B;
        I++;
    }
    cout << "Potencia = " << P << endl;
    return 0;
}

```

Exercício 2G - do...while

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int B, E, P, I;
    cout << "Entre o valor da base .....: "; cin >> B;
    cout << "Entre o valor da expoente ...: "; cin >> E;

```

```

P = 1;
I = 1;
do
{
    P *= B;
    I++;
}
while (I <= E);
cout << "Potencia = " << P << endl;
return 0;
}

```

Exercício 3G - for()

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int B, E, P, I;
    cout << "Entre o valor da base .....: "; cin >> B;
    cout << "Entre o valor da expoente ...: "; cin >> E;
    P = 1;
    for (I = 1; I <= E; I++)
        P *= B;
    cout << "Potencia = " << P << endl;
    return 0;
}

```

Capítulo 5

Exercício 1A - Função como procedimento

```

#include <iostream>
#include <iomanip>
using namespace std;

int I, S, N;

int somatorio(void);

int main(void)
{
    cin >> N;
    somatorio();
    return 0;
}

int somatorio(void)
{
    S = 0;
    for (I = 1; I <= N; I++)
        S += I;
    cout << "Soma = " << setw(5) << S << endl;
    return 0;
}

```

Exercício 1B - Função como procedimento

```

#include <iostream>
#include <iomanip>
using namespace std;

```

```

float VALOR, TAXA, TEMPO, PREST;

int prestacao(void);

int main(void)
{
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cout << "Valor .....: "; cin >> VALOR;
    cout << "Taxa .....: "; cin >> TAXA;
    cout << "Tempo .....: "; cin >> TEMPO;
    prestacao();
    return 0;
}

int prestacao(void)
{
    PREST = VALOR + (VALOR * (TAXA/100) * TEMPO);
    cout << "Prestacao = " << setw(5) << PREST << endl;
    return 0;
}

```

Exercício 1C - Função como procedimento

```

#include <iostream>
#include <iomanip>
using namespace std;

int I, P, B, E;

int potencia(void);

int main(void)
{
    cout << "Entre a base .....: "; cin >> B;
    cout << "Entre a expoente ....: "; cin >> E;
    potencia();
    return 0;
}

int potencia(void)
{
    P = 1;
    for (I = 1; I <= E; I++)
        P *= B;
    cout << "Resultado = " << setw(5) << P << endl;
    return 0;
}

```

Exercício 1D - Função como procedimento

```

#include <iostream>
#include <iomanip>
using namespace std;

float C, F;

int temperatura(void);

int main(void)
{
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> C;
}

```



```

    temperatura();
    return 0;
}

int temperatura(void)
{
    F = (9 * C + 160) / 5;
    cout << setw(5) << F << endl;
    return 0;
}

```

Exercício 1E - Função como procedimento

```

#include <iostream>
#include <iomanip>
using namespace std;

float F, C;

int temperatura(void);

int main(void)
{
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> F;
    temperatura();
    return 0;
}

int temperatura(void)
{
    C = (F - 32) * 5 / 9;
    cout << setw(5) << C << endl;
    return 0;
}

```

Exercício 1F - Função como procedimento

```

#include <iostream>
#include <iomanip>
using namespace std;

float VOLUME, COMPRIMENTO, LARGURA, ALTURA;

int caixa(void);

int main(void)
{
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cout << "Comprimento .. "; cin >> COMPRIMENTO;
    cout << "Largura ..... "; cin >> LARGURA;
    cout << "Altura ..... "; cin >> ALTURA;
    caixa();
    return 0;
}

int caixa(void)
{
    VOLUME = COMPRIMENTO * LARGURA * ALTURA;
    cout << "Volume = " << setw(5) << VOLUME << endl;
    return 0;
}

```

Exercício 2A - Função: pass. par. por valor

```
#include <iostream>
#include <iomanip>
using namespace std;

int somatorio(int X);

int main(void)
{
    int N;
    cin >> N;
    somatorio(N);
    return 0;
}

int somatorio(int X)
{
    int I, S;
    S = 0;
    for (I = 1; I <= X; I++)
        S += I;
    cout << "Soma = " << setw(5) << S << endl;
    return 0;
}
```

Exercício 2B - Função: pass. par. por valor

```
#include <iostream>
#include <iomanip>
using namespace std;

int prestacao(float VALOR, float TAXA, float TEMPO);

int main(void)
{
    float VLR, TX, TMP;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cout << "Valor .....: "; cin >> VLR;
    cout << "Taxa .....: "; cin >> TX;
    cout << "Tempo .....: "; cin >> TMP;
    prestacao(VLR, TX, TMP);
    return 0;
}

int prestacao(float VALOR, float TAXA, float TEMPO)
{
    float PREST;
    PREST = VALOR + (VALOR * (TAXA/100) * TEMPO);
    cout << "Prestacao = " << setw(5) << PREST << endl;
    return 0;
}
```

Exercício 2C - Função: pass. par. por valor

```
#include <iostream>
#include <iomanip>
using namespace std;

int potencia(int B, int E);

int main(void)
{

```

```

    int B, E;
    cout << "Entre a base .....: "; cin >> B;
    cout << "Entre a expoente ...: "; cin >> E;
    potencia(B, E);
    return 0;
}

int potencia(int B, int E)
{
    int I, P;
    P = 1;
    for (I = 1; I <= E; I++)
        P *= B;
    cout << "Resultado = " << setw(5) << P << endl;
    return 0;
}

```

Exercício 2D - Função: pass. par. por valor

```

#include <iostream>
#include <iomanip>
using namespace std;

int temperatura(float C);

int main(void)
{
    float C;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> C;
    temperatura(C);
    return 0;
}

int temperatura(float C)
{
    float F;
    F = (9 * C + 160) / 5;
    cout << setw(5) << F << endl;
    return 0;
}

```

Exercício 2E - Função: pass. par. por valor

```

#include <iostream>
#include <iomanip>
using namespace std;

int temperatura(float F);

int main(void)
{
    float F;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> F;
    temperatura(F);
    return 0;
}

int temperatura(float F)
{
    float C;
    C = (F - 32) * 5 / 9;
}

```

```

    cout << setw(5) << C << endl;
    return 0;
}

```

Exercício 2F - Função: pass. par. por valor

```

#include <iostream>
#include <iomanip>
using namespace std;

int caixa(float COMPRIMENTO, float LARGURA, float ALTURA);

int main(void)
{
    float COMPRIMENTO, LARGURA, ALTURA;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cout << "Comprimento ..: "; cin >> COMPRIMENTO;
    cout << "Largura .....: "; cin >> LARGURA;
    cout << "Altura .....: "; cin >> ALTURA;
    caixa(COMPRIMENTO, LARGURA, ALTURA);
    return 0;
}

int caixa(float COMPRIMENTO, float LARGURA, float ALTURA)
{
    float VOLUME;
    VOLUME = COMPRIMENTO * LARGURA * ALTURA;
    cout << "Volume = " << setw(5) << VOLUME << endl;
    return 0;
}

```

Exercício 3A - Função: pass. par. por referência

```

#include <iostream>
#include <iomanip>
using namespace std;

int somatorio(int X, int *Y);

int main(void)
{
    int N, X;
    cin >> N;
    somatorio(N, &X);
    cout << "Soma = " << setw(5) << X << endl;
    return 0;
}

int somatorio(int X, int *S)
{
    int I;
    *S = 0;
    for (I = 1; I <= X; I++)
        *S += I;
    return 0;
}

```

Exercício 3B - Função: pass. par. por referência

```

#include <iostream>
#include <iomanip>
using namespace std;

int prestacao(float VALOR, float TAXA, float TEMPO, float *PREST);

```

```

int main(void)
{
    float VLR, TX, TMP, P;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cout << "Valor .....: "; cin >> VLR;
    cout << "Taxa .....: "; cin >> TX;
    cout << "Tempo .....: "; cin >> TMP;
    prestacao(VLR, TX, TMP, &P);
    cout << "Prestacao = " << setw(5) << P << endl;
    return 0;
}

int prestacao(float VALOR, float TAXA, float TEMPO, float *PREST)
{
    *PREST = VALOR + (VALOR * (TAXA/100) * TEMPO);
    return 0;
}

```

Exercício 3C - Função: pass. par. por referência

```

#include <iostream>
#include <iomanip>
using namespace std;

int potencia(int B, int E, int *P);

int main(void)
{
    int B, E, R;
    cout << "Entre a base .....: "; cin >> B;
    cout << "Entre a expoente ...: "; cin >> E;
    potencia(B, E, &R);
    cout << "Resultado = " << setw(5) << R << endl;
    return 0;
}

int potencia(int B, int E, int *P)
{
    int I;
    *P = 1;
    for (I = 1; I <= E; I++)
        *P *= B;
    return 0;
}

```

Exercício 3D - Função: pass. par. por referência

```

#include <iostream>
#include <iomanip>
using namespace std;

int temperatura(float C, float *F);

int main(void)
{
    float C, R;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> C;
    temperatura(C, &R);
    cout << setw(5) << R << endl;
    return 0;
}

```

```
int temperatura(float C, float *F)
{
    *F = (9 * C + 160) / 5;
    return 0;
}
```

Exercício 3E - Função: pass. par. por referência

```
#include <iostream>
#include <iomanip>
using namespace std;

int temperatura(float F, float *C);

int main(void)
{
    float F, R;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> F;
    temperatura(F, &R);
    cout << setw(5) << R << endl;
    return 0;
}

int temperatura(float F, float *C)
{
    *C = (F - 32) * 5 / 9;
    return 0;
}
```

Exercício 3F - Função: pass. par. por referência

```
#include <iostream>
#include <iomanip>
using namespace std;

int caixa(float COMPRIMENTO, float LARGURA, float ALTURA, float *VOLUME);

int main(void)
{
    float COMPRIMENTO, LARGURA, ALTURA, VOLUME;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cout << "Comprimento ..: "; cin >> COMPRIMENTO;
    cout << "Largura .....: "; cin >> LARGURA;
    cout << "Altura .....: "; cin >> ALTURA;
    caixa(COMPRIMENTO, LARGURA, ALTURA, &VOLUME);
    cout << "Volume = " << setw(5) << VOLUME << endl;
    return 0;
}

int caixa(float COMPRIMENTO, float LARGURA, float ALTURA, float *VOLUME)
{
    *VOLUME = COMPRIMENTO * LARGURA * ALTURA;
    return 0;
}
```

Exercício 4A - Função com retorno de valor

```
#include <iostream>
#include <iomanip>
using namespace std;
```

```

int somatorio(int X);

int main(void)
{
    int N;
    cin >> N;
    cout << "Soma = " << setw(5) << somatorio(N) << endl;
    return 0;
}

int somatorio(int X)
{
    int I, S;
    S = 0;
    for (I = 1; I <= X; I++)
        S += I;
    return S;
}

```

Exercício 4B - Função com retorno de valor

```

#include <iostream>
#include <iomanip>
using namespace std;

float prestacao(float VALOR, float TAXA, float TEMPO);

int main(void)
{
    float VLR, TX, TMP;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cout << "Valor .....: "; cin >> VLR;
    cout << "Taxa .....: "; cin >> TX;
    cout << "Tempo .....: "; cin >> TMP;
    cout << "Prestacao = " << setw(5) << prestacao(VLR, TX, TMP) << endl;
    return 0;
}

float prestacao(float VALOR, float TAXA, float TEMPO)
{
    float PREST;
    PREST = VALOR + (VALOR * (TAXA/100) * TEMPO);
    return PREST;
}

```

Exercício 4C - Função com retorno de valor

```

#include <iostream>
#include <iomanip>
using namespace std;

int potencia(int B, int E);

int main(void)
{
    int B, E;
    cout << "Entre a base .....: "; cin >> B;
    cout << "Entre a expoente ...: "; cin >> E;
    cout << "Resultado = " << setw(5) << potencia(B, E) << endl;
    return 0;
}

int potencia(int B, int E)
{

```

```

int I, P;
P = 1;
for (I = 1; I <= E; I++)
    P *= B;
return P;
}

```

Exercício 4D - Função com retorno de valor

```

#include <iostream>
#include <iomanip>
using namespace std;

float temperatura(float C);

int main(void)
{
    float C;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> C;
    cout << setw(5) << temperatura(C) << endl;
    return 0;
}

float temperatura(float C)
{
    float F;
    F = (9 * C + 160) / 5;
    return F;
}

```

Exercício 4E - Função com retorno de valor

```

#include <iostream>
#include <iomanip>
using namespace std;

float temperatura(float F);

int main(void)
{
    float F;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cin >> F;
    cout << setw(5) << temperatura(F) << endl;
    return 0;
}

float temperatura(float F)
{
    float C;
    C = (F - 32) * 5 / 9;
    return C;
}

```

Exercício 4F - Função com retorno de valor

```

#include <iostream>
#include <iomanip>
using namespace std;

float caixa(float COMPRIMENTO, float LARGURA, float ALTURA);

```



```

int main(void)
{
    float COMPRIMENTO, LARGURA, ALTURA;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    cout << "Comprimento ..: "; cin >> COMPRIMENTO;
    cout << "Largura .....: "; cin >> LARGURA;
    cout << "Altura .....: "; cin >> ALTURA;
    cout << "Volume = " << setw(5) << caixa(COMPRIMENTO, LARGURA, ALTURA) << endl;
    return 0;
}

float caixa(float COMPRIMENTO, float LARGURA, float ALTURA)
{
    float VOLUME;
    VOLUME = COMPRIMENTO * LARGURA * ALTURA;
    return VOLUME;
}

```

Capítulo 6

Exercício 1A

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[10], I;
    cout << setiosflags(ios::right);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 9; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << ": ";
        cin >> A[I];
    }

    // Apresentacao dos dados da matriz "A"

    for (I = 0; I <= 9; I++)
        cout << "\nA[" << setw(2) << I + 1 << "] = " << setw(3) << A[I];
    return 0;
}

```

Exercício 1B

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[8], B[8], I;
    cout << setiosflags(ios::right);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 7; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << ": ";

```

```

        cin >> A[I];
    }

    // Processamento dos dados da matriz "B"

    for (I = 0; I <= 7; I++)
        B[I] = A[I] * 3;

    // Apresentacao dos dados da matriz "B"

    for (I = 0; I <= 7; I++)
        cout << "\nB[" << setw(2) << I + 1 << "] = " << setw(3) << B[I];
    return 0;
}

```

Exercício 1C

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[8], B[8], I, J;
    cout << setiosflags(ios::right);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 7; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << ": ";
        cin >> A[I];
    }

    // Processamento dos dados da matriz "B"

    for (I = 0; I <= 7; I++)
    {
        B[I] = 1;
        for (J = 1; J <= A[I]; J++)
            B[I] *= J;
    }

    // Apresentacao dos dados da matriz "B"

    for (I = 0; I <= 7; I++)
        cout << "\nB[" << setw(2) << I + 1 << "] = " << setw(5) << B[I];
    return 0;
}

```

Exercício 1D

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[8], B[8], C[8], I;
    cout << setiosflags(ios::right);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 7; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << " da matriz 'A': ";

```

```

        cin >> A[I];
    }

    // Entrada de dados na matriz "B"

    for (I = 0; I <= 7; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << " da matriz
'B': ";
        cin >> B[I];
    }

    // Processamento dos dados da matriz "C"

    for (I = 0; I <= 7; I++)
        C[I] = A[I] - B[I];

    // Apresentacao dos dados da matriz "C"

    for (I = 0; I <= 7; I++)
        cout << "\nC[" << setw(2) << I + 1 << "] = " << setw(5) << C[I];
    return 0;
}

```

Exercício 1E

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[10], B[10], C[20], I;
    cout << setiosflags(ios::right);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 9; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << " da matriz
'A': ";
        cin >> A[I];
    }

    // Entrada de dados na matriz "B"

    for (I = 0; I <= 9; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << " da matriz
'B': ";
        cin >> B[I];
    }

    // Processamento dos dados da matriz "C"

    for (I = 0; I <= 19; I++)
        if (I <= 9)
            C[I] = A[I];
        else
            C[I] = B[I - 10];

    // Apresentacao dos dados da matriz "C"

    for (I = 0; I <= 19; I++)
        cout << "\nC[" << setw(2) << I + 1 << "] = " << setw(5) << C[I];
    return 0;
}

```

Exercício 1F

```
#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[5], B[10], C[15], I, R;
    cout << setiosflags(ios::right);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 4; I++)
    {
        do {
            cout << "Informe um valor 'par' para o elemento nr. " << setw(2) << I + 1 << " da
matriz 'A' .....: ";
            cin >> A[I];
            R = A[I] % 2;
        } while (!(R == 0));
    }

    // Entrada de dados na matriz "B"

    for (I = 0; I <= 9; I++)
    {
        do {
            cout << "Informe um valor 'impar' para o elemento nr. " << setw(2) << I + 1 << "
da matriz 'B' ....: ";
            cin >> B[I];
            R = B[I] % 2;
        } while (!(R != 0));
    }

    // Processamento dos dados da matriz "C"

    for (I = 0; I <= 14; I++)
        if (I <= 4)
            C[I] = A[I];
        else
            C[I] = B[I - 5];

    // Apresentacao dos dados da matriz "C"

    for (I = 0; I <= 14; I++)
        cout << "\nC[" << setw(2) << I + 1 << "] = " << setw(5) << C[I];
    return 0;
}
```

Exercício 1G

```
#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[15], B[15], I;
    cout << setiosflags(ios::right);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 14; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << ": ";
    }
```

```

        cin >> A[I];
    }

    // Processamento dos dados da matriz "B"

    for (I = 0; I <= 14; I++)
        B[I] = A[I] * A[I];

    // Apresentacao dos dados da matriz "B"

    for (I = 0; I <= 14; I++)
        cout << "\nB[" << setw(2) << I + 1 << "] = " << setw(3) << B[I];
    return 0;
}

```

Exercício 1H

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[10], B[10], I;
    cout << setiosflags(ios::right);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 9; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << ": ";
        cin >> A[I];
    }

    // Processamento da inversao dos valores entre as matrizes "A" e "B"

    for (I = 0; I <= 9; I++)
        B[I] = A[9 - I];

    // Apresentacao dos dados da matriz "B"

    for (I = 0; I <= 9; I++)
    {
        cout << "\nA[" << setw(2) << I + 1 << "] = " << setw(3) << A[I];
        cout << " | B[" << setw(2) << I + 1 << "] = " << setw(3) << B[I];
    }
    return 0;
}

```

Exercício 2A

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[5][3], B[5][3], C[5][3], I, J;
    cout << setiosflags(ios::right);

    // Entrada de dados

    cout << "\nInforme dados da matriz A" << endl;
    for (I = 0; I <= 4; I++)
    {
        cout << "\nColuna ...: " << I + 1 << endl;
        for (J = 0; J <= 2; J++)

```

```

        {
            cout << "Linha ....: " << J + 1 << " -> ";
            cin >> A[I][J];
        }
    }

    cout << "\nInforme dados da matriz B" << endl;
    for (I = 0; I <= 4; I++)
    {
        cout << "\nColuna ...: " << I + 1 << endl;
        for (J = 0; J <= 2; J++)
        {
            cout << "Linha ....: " << J + 1 << " -> ";
            cin >> B[I][J];
        }
    }

    // Processamento

    for (I = 0; I <= 4; I++)
        for (J = 0; J <= 2; J++)
            C[I][J] = A[I][J] + B[I][J];

    // Saida dos dados

    cout << "\nResultados" << endl;
    for (I = 0; I <= 4; I++)
    {
        cout << "\nColuna ...: " << I + 1;
        for (J = 0; J <= 2; J++)
            cout << "\nLinha ....: " << J + 1 << " -> " << C[I][J];
        cout << endl;
    }

    return 0;
}

```

Exercício 2B

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[5], B[5][3], I, J;
    cout << setiosflags(ios::right);

    // Entrada de dados

    cout << "\nInforme dados da matriz A" << endl;
    for (I = 0; I <= 4; I++)
    {
        cout << "A[" << I + 1 << "] = ";
        cin >> A[I];
    }

    // Processamento

    for (I = 0; I <= 4; I++)
    {
        B[I][0] = A[I] + 5;

        B[I][1] = 1;
        for (J = 1; J <= A[I]; J++)
            B[I][1] *= J;
    }
}

```

```

        B[I][2] = A[I] * A[I];
    }

    // Saida dos dados

    cout << "\nResultados" << endl;
    for (I = 0; I <= 4; I++)
    {
        cout << "B[" << I + 1 << ",1] = " << setw(6) << B[I][0] << " | ";
        cout << "B[" << I + 1 << ",2] = " << setw(6) << B[I][1] << " | ";
        cout << "B[" << I + 1 << ",3] = " << setw(6) << B[I][2] << endl;
    }

    return 0;
}

```

Exercício 2C

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    float A[4][4], B[4][4], C[4][4];
    int I, J;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);

    // Entrada de dados

    cout << "\nInforme dados da matriz A" << endl;
    for (I = 0; I <= 3; I++)
    {
        cout << "\nColuna ...: " << I + 1 << endl;
        for (J = 0; J <= 3; J++)
        {
            cout << "Linha ....: " << J + 1 << " -> ";
            cin >> A[I][J];
        }
    }

    cout << "\nInforme dados da matriz B" << endl;
    for (I = 0; I <= 3; I++)
    {
        cout << "\nColuna ...: " << I + 1 << endl;
        for (J = 0; J <= 3; J++)
        {
            cout << "Linha ....: " << J + 1 << " -> ";
            cin >> B[I][J];
        }
    }

    // Processamento

    for (I = 0; I <= 3; I++)
        for (J = 0; J <= 3; J++)
            C[I][J] = A[I][J] - B[I][J];

    // Saida dos dados

    cout << "\nResultados" << endl;
    for (I = 0; I <= 3; I++)
    {
        cout << "\nColuna ...: " << I + 1;
    }
}

```

```

        for (J = 0; J <= 3; J++)
            cout << "\nLinha ....: " << J + 1 << " -> " << setw(8) << C[I][J];
        cout << endl;
    }

    return 0;
}

```

Exercício 2D

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    float A[4][4];
    int I, J;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);

    // Entrada de dados

    cout << "\nInforme dados da matriz A" << endl;
    for (I = 0; I <= 3; I++)
    {
        cout << "\nColuna ....: " << I + 1 << endl;
        for (J = 0; J <= 3; J++)
        {
            cout << "Linha ....: " << J + 1 << " -> ";
            cin >> A[I][J];
        }
    }

    // Processamento e saída dos elementos da diagonal principal

    cout << "\nResultado" << endl;
    for (I = 0; I <= 3; I++)
        for (J = 0; J <= 3; J++)
            if (I == J)
                cout << "Dado: " << setw(8) << A[I][J] << endl;
    cout << endl;

    return 0;
}

```

Exercício 2E

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    float A[3][3];
    int I, J;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);

    // Entrada de dados

    cout << "\nInforme dados da matriz A" << endl;
    for (I = 0; I <= 2; I++)
    {
        cout << "\nColuna ....: " << I + 1 << endl;

```



```

        for (J = 0; J <= 2; J++)
        {
            cout << "Linha ....: " << J + 1 << " -> ";
            cin >> A[I][J];
        }
    }

    // Processamento e saida

    cout << "\nResultado" << endl;
    for (I = 0; I <= 2; I++)
        for (J = 0; J <= 2; J++)
            if (I == J)
                cout << "Dado: " << setw(8) << A[I][J] * 2 << endl;
            else
                cout << "Dado: " << setw(8) << A[I][J] * 3 << endl;
            cout << endl;

    return 0;
}

```

Exercício 3A

```

#include <iostream>
#include <iomanip>
#include <cstring>
using namespace std;

int main(void)
{
    int I, J;
    char NOME[12][40], X[40];
    cout << "Leitura e apresentacao de nomes\n\n";

    // Entrada dos nomes

    for (I = 0; I <= 11; I++)
    {
        cout << "Informe o " << setw(2) << I + 1 << "o. nome: ";
        cin.getline(NOME[I], sizeof(NOME[I]));
    }
    cout << endl;

    // Classificacao dos nomes

    for (I = 0; I <= 10; I++)
        for (J = I + 1; J <= 11; J++)
            if (strcmp(NOME[I], NOME[J]) < 0)
            {
                strcpy(X, NOME[I]);
                strcpy(NOME[I], NOME[J]);
                strcpy(NOME[J], X);
            }

    // Apresentacao dos nomes

    for (I = 0; I <= 11; I++)
        cout << setw(2) << I + 1 << " nome: " << NOME[I] << endl;

    return 0;
}

```

Exercício 3B

```

#include <iostream>
#include <iomanip>
#include <cctype>

```

```

using namespace std;

int main(void)
{
    int A[8], B[8], I, J, X, COMECO, FINAL, MEIO, PESQ;
    bool ACHA;
    char RESP;
    cout << setiosflags(ios::right);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 7; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << ": ";
        cin >> A[I];
    }

    // Processamento dos dados da matriz "B"

    for (I = 0; I <= 7; I++)
        B[I] = A[I] * 5;

    // Classificacao dos valores da matriz B

    for (I = 0; I <= 7; I++)
        for (J = I + 1; J <= 8; J++)
            if (B[I] > B[J])
            {
                X = B[I];
                B[I] = B[J];
                B[J] = X;
            }

    // Apresentacao dos dados ordenados da matriz "B"

    for (I = 0; I <= 7; I++)
        cout << "\nB[" << setw(2) << I + 1 << "] = " << setw(3) << B[I];
    cout << endl;

    // Pesquisa binaria de conteudo

    do
    {
        cout << "\nEntre o valor a ser pesquisado: "; cin >> PESQ;
        COMECO = 0;
        FINAL = 9;
        ACHA = false;
        while (COMECO <= FINAL && ACHA == false)
        {
            MEIO = (COMECO + FINAL) / 2;
            if (PESQ == B[MEIO])
                ACHA = true;
            else
            {
                if (PESQ < B[MEIO])
                    FINAL = MEIO - 1;
                else
                    COMECO = MEIO + 1;
            }
        }
        if (ACHA == true)
            cout << PESQ << " foi localizado na posicao " << MEIO + 1;
        else
            cout << PESQ << " nao foi localizado";
        cout << endl;
        cout << "\n\nContinua? [S]IM/[N]AO + <Enter>: "; cin >> RESP;
        RESP = toupper(RESP);
    }
    while (RESP == 'S');
}

```

```

    return 0;
}

```

Exercício 3C

```

#include <iostream>
#include <iomanip>
using namespace std;

int main(void)
{
    int A[10], B[10], I, J, X;
    cout << setiosflags(ios::right);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 9; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << ": ";
        cin >> A[I];
    }

    // Processamento dos dados da matriz "B"

    for (I = 0; I <= 9; I++)
    {
        B[I] = 1;
        for (J = 1; J <= A[I]; J++)
            B[I] *= J;
    }

    // Classificacao dos valores da matriz "B"

    for (I = 0; I <= 8; I++)
        for (J = I + 1; J <= 9; J++)
            if (B[I] > B[J])
            {
                X = B[I];
                B[I] = B[J];
                B[J] = X;
            }

    // Apresentacao dos dados da matriz "B"

    for (I = 0; I <= 9; I++)
        cout << "\nB[" << setw(2) << I + 1 << "] = " << setw(7) << B[I];
    return 0;
}

```

Exercício 3D

```

#include <iostream>
#include <iomanip>
#include <cstring>
using namespace std;

int main(void)
{
    int I, J;
    char A[12][40], B[11][40], C[23][40], X[40];
    cout << "Leitura e apresentacao de nomes\n\n";

    // Entrada dos nomes

    for (I = 0; I <= 11; I++)
    {
        cout << "Informe o " << setw(2) << I + 1 << "o. nome feminino .... ";
    }
}

```

```

        cin.getline(A[I], sizeof(A[I]));
    }
    cout << endl;

    for (I = 0; I <= 10; I++)
    {
        cout << "Informe o " << setw(2) << I + 1 << "o. nome masculino ... ";
        cin.getline(B[I], sizeof(B[I]));
    }
    cout << endl;

    // Juncao - Matriz C

    for (I = 0; I <= 22; I++)
        if (I <= 11)
            strcpy(C[I], A[I]);
        else
            strcpy(C[I], B[I - 12]);

    // Ordenacao de strings

    for (I = 0; I <= 21; I++)
        for (J = I + 1; J <= 22; J++)
            if (strcmp(C[I], C[J]) > 0)
            {
                strcpy(X, C[I]);
                strcpy(C[I], C[J]);
                strcpy(C[J], X);
            }

    // Apresentacao dos nomes

    for (I = 0; I <= 22; I++)
        cout << setw(2) << I + 1 << " nome: " << C[I] << endl;

    return 0;
}

```

Exercício 3E

```

#include <iostream>
#include <iomanip>
#include <cctype>
using namespace std;

int main(void)
{
    float A[10], B[10], PESQ;
    int I, J;
    bool ACHA;
    char RESP;
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);

    // Entrada de dados na matriz "A"

    for (I = 0; I <= 9; I++)
    {
        cout << "Informe um valor para o elemento nr. " << setw(2) << I + 1 << ": ";
        cin >> A[I];
    }

    // Processamento dos dados da matriz "B"

    for (I = 0; I <= 9; I++)
        B[I] = A[I] + 2;
}

```

```

// Apresentacao dos dados por pesquisa sequencial

do
{
    cout << "\nEntre o valor a ser pesquisado: "; cin >> PESQ;
    I = 0;
    ACHA = false;
    while (I <= 9 && ACHA == false)
        if (PESQ == B[I])
            ACHA = true;
        else
            I++;
    if (ACHA == true)
        cout << PESQ << " foi localizado na posicao " << I + 1;
    else
        cout << PESQ << " nao foi localizado";
    cout << endl;
    cout << "\n\nContinua? [S]IM/[N]AO + <Enter>: "; cin >> RESP;
    RESP = toupper(RESP);
}
while (RESP == 'S');

return 0;
}

```

Capítulo 11

Exercício 1A

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

int main(void)
{
    int I, A[10];
    ofstream ARQBIN;
    ARQBIN.open("DADOS11A.DBC", ios_base::binary);
    for (I = 0; I < 10; I++)
    {
        cout << "Digite o elemento " << setw(2) << I + 1 << " - ";
        cin >> A[I];
    }
    ARQBIN.write(reinterpret_cast<char*>(&A), sizeof(A));
    ARQBIN.close();
    return 0;
}

```

Exercício 1B

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

int main(void)
{
    int I, A[10];
    ifstream ARQBIN;
    ARQBIN.open("DADOS11A.DBC", ios_base::binary);
    ARQBIN.read(reinterpret_cast<char*>(&A), sizeof(A));
    for (I = 0; I < 10; I++)

```

```

    {
        cout << "Elemento " << setw(2) << I + 1 << " = ";
        cout << A[I] << endl;
    }
    ARQBIN.close();
    return 0;
}

```

Exercício 1C

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

int main(void)
{
    int R, I, A[10];
    ifstream ARQBIN;
    ARQBIN.open("DADOS11A.DBC", ios_base::binary);
    ARQBIN.read(reinterpret_cast<char*>(&A), sizeof(A));
    for (I = 0; I < 10; I++)
    {
        R = A[I] % 2;
        if (R == 0)
        {
            cout << "Posicao = " << setw(2) << I + 1 << " | Elemento = ";
            cout << A[I] << endl;
        }
    }
    ARQBIN.close();
    return 0;
}

```

Exercício 1D

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

int main(void)
{
    int R, I, A[10];
    ifstream ARQBIN;
    ARQBIN.open("DADOS11A.DBC", ios_base::binary);
    ARQBIN.read(reinterpret_cast<char*>(&A), sizeof(A));
    for (I = 0; I < 10; I++)
    {
        R = A[I] % 2;
        if (R != 0)
        {
            cout << "Posicao = " << setw(2) << I + 1 << " | Elemento = ";
            cout << A[I] << endl;
        }
    }
    ARQBIN.close();
    return 0;
}

```

Exercício 1E

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

```

```

int main(void)
{
    int R2, R3, I, A[10];
    ifstream ARQBIN;
    ARQBIN.open("DADOS11A.DBC", ios_base::binary);
    ARQBIN.read(reinterpret_cast<char*>(&A), sizeof(A));
    for (I = 0; I < 10; I++)
    {
        R2 = A[I] % 2;
        R3 = A[I] % 3;
        if (R2 == 0 and R3 == 0)
        {
            cout << "Posicao = " << setw(2) << I + 1 << " | Elemento = ";
            cout << A[I] << endl;
        }
    }
    ARQBIN.close();
    return 0;
}

```

Exercício 1F

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

int main(void)
{
    int I, A[10], B[10], C[10];

    ofstream ARQBIN1;
    ARQBIN1.open("DADOS11F.DBC", ios_base::binary);
    for (I = 0; I < 10; I++)
    {
        cout << "Digite o elemento " << setw(2) << I + 1 << " para a matriz 'A' - ";
        cin >> A[I];
    }
    for (I = 0; I < 10; I++)
        B[I] = A[I] * 2;
    ARQBIN1.write(reinterpret_cast<char*>(&B), sizeof(B));
    ARQBIN1.close();

    ifstream ARQBIN2;
    ARQBIN2.open("DADOS11F.DBC", ios_base::binary);
    ARQBIN2.read(reinterpret_cast<char*>(&C), sizeof(C));
    for (I = 0; I < 10; I++)
    {
        cout << "Elemento " << setw(2) << I + 1 << " = ";
        cout << C[I] << endl;
    }
    ARQBIN2.close();

    return 0;
}

```

Exercício 1G

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

int main(void)
{
    int R, I, S = 0, X[10];

```

```

ifstream ARQBIN;
ARQBIN.open("DADOS11F.DBC", ios_base::binary);
ARQBIN.read(reinterpret_cast<char*>(&X), sizeof(X));
for (I = 0; I < 10; I++)
{
    R = X[I] % 2;
    if (R == 0)
        S += X[I];
}
cout << "Soma = " << S << endl;
ARQBIN.close();

return 0;
}

```

Exercício 1H

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

int main(void)
{
    int I;
    float A[8], B[8];
    ofstream ARQBIN1("ARQ1.DBC", ios_base::binary);
    ofstream ARQBIN2("ARQ2.DBC", ios_base::binary);
    for (I = 0; I < 8; I++)
    {
        cout << "Digite o elemento " << setw(2) << I + 1 << " da matriz A - ";
        cin >> A[I];
    }
    for (I = 0; I < 8; I++)
    {
        cout << "Digite o elemento " << setw(2) << I + 1 << " da matriz B - ";
        cin >> B[I];
    }
    ARQBIN1.write(reinterpret_cast<char*>(&A), sizeof(A));
    ARQBIN2.write(reinterpret_cast<char*>(&B), sizeof(B));
    ARQBIN1.close();
    ARQBIN2.close();
    return 0;
}

```

Exercício 1I

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

int main(void)
{
    int I;
    float X[8], Y[8], Z[8];
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    ifstream ARQBIN1("ARQ1.DBC", ios_base::binary);
    ifstream ARQBIN2("ARQ2.DBC", ios_base::binary);
    ARQBIN1.read(reinterpret_cast<char*>(&X), sizeof(X));
    ARQBIN2.read(reinterpret_cast<char*>(&Y), sizeof(Y));
    for (I = 0; I < 8; I++)
        Z[I] = X[I] + Y[I];
    for (I = 0; I < 8; I++)

```



```

        cout << setw(8) << Z[I] << endl;
    ARQBIN1.close();
    ARQBIN2.close();
    return 0;
}

```

Exercício 1J

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

int main(void)
{
    int I;
    float A[8], B[8], C[16];
    cout << setprecision(2);
    cout << setiosflags(ios::right);
    cout << setiosflags(ios::fixed);
    ifstream ARQBIN1("ARQ1.DBC", ios_base::binary);
    ifstream ARQBIN2("ARQ2.DBC", ios_base::binary);
    ofstream ARQBIN3("ARQ3.DBC", ios_base::binary);
    ARQBIN1.read(reinterpret_cast<char*>(&A), sizeof(A));
    ARQBIN2.read(reinterpret_cast<char*>(&B), sizeof(B));
    for (I = 0; I <= 15; I++)
        if (I <= 7)
            C[I] = A[I];
        else
            C[I] = B[I - 8];
    ARQBIN3.write(reinterpret_cast<char*>(&C), sizeof(C));
    ARQBIN1.close();
    ARQBIN2.close();
    ARQBIN3.close();
    return 0;
}

```

Exercício 1K

```

#include <iostream>
#include <iomanip>
#include <fstream>
using namespace std;

int main(void)
{
    int I;
    float A[16];
    ifstream ARQBIN("ARQ3.DBC", ios_base::binary);
    ARQBIN.read(reinterpret_cast<char*>(&A), sizeof(A));
    for (I = 0; I < 16; I++)
    {
        cout << "Elemento " << setw(2) << I + 1 << " = ";
        cout << A[I] << endl;
    }
    ARQBIN.close();
    return 0;
}

```