

## ● Lista de Exercícios PUCRS

1. Escrever um algoritmo que lê 5 valores para a, um de cada vez, e conta quantos destes valores são negativos, escrevendo esta informação.

```
#include <stdio.h>

int main() {
    int a, i, count = 0;
    for(i = 0; i < 5; i++) {
        printf("Digite um número: ");
        scanf("%d", &a);
        if(a < 0) {
            count++;
        }
    }
    printf("Número de valores negativos: %d\n", count);
    return 0;
}
```

2. Escrever um algoritmo que lê um valor N inteiro e positivo e que calcula e escreve o valor de E.  
 $E = 1 + 1 / 1! + 1 / 2! + 1 / 3! + 1 / N!$

```
#include <stdio.h>

double fatorial(int n) {
    double f = 1;
    for(int i = 1; i <= n; i++) {
        f *= i;
    }
    return f;
}

double calcular_E(int n) {
    double E = 1.0;
    for(int i = 1; i <= n; i++) {
        E += 1 / fatorial(i);
    }
    return E;
}

int main() {
    int n;
    printf("Digite um número inteiro e positivo: ");
    scanf("%d", &n);
    printf("O valor de E é: %f\n", calcular_E(n));
    return 0;
}
```

3. A prefeitura de uma cidade fez uma pesquisa entre seus habitantes, coletando dados sobre o salário e número de filhos. A prefeitura deseja saber: a) média do salário da população; b) média do número de filhos; c) maior salário; d) percentual de pessoas com salário até R\$100,00. O final da leitura de dados se dará com a entrada de um salário negativo. (Use o comando ENQUANTO-FAÇA).

```
#include <stdio.h>

int main() {
    double salario, total_salario = 0, maior_salario = 0;
    int filhos, total_filhos = 0, cont = 0, cont_100 = 0;

    printf("Digite o salário: ");
    scanf("%lf", &salario);

    while(salario >= 0) {
        printf("Digite o número de filhos: ");
        scanf("%d", &filhos);

        total_salario += salario;
        total_filhos += filhos;
        cont++;

        if(salario > maior_salario) {
            maior_salario = salario;
        }

        if(salario <= 100) {
            cont_100++;
        }

        printf("Digite o salário: ");
        scanf("%lf", &salario);
    }

    printf("Média do salário da população: %.2lf\n", total_salario / cont);
    printf("Média do número de filhos: %.2lf\n", (double)total_filhos / cont);
    printf("Maior salário: %.2lf\n", maior_salario);
    printf("Percentual de pessoas com salário até R$100,00: %.2lf%%\n", (double)cont_100 / cont * 100);

    return 0;
}
```

4. Chico tem 1,50 metro e cresce 2 centímetros por ano, enquanto Zé tem 1,10 metro e cresce 3 centímetros por ano. Construa um algoritmo que calcule e imprima quantos anos serão necessários para que Zé seja maior que Chico.

```
#include <stdio.h>

int main() {
    float chico = 1.50, ze = 1.10;
    int anos = 0;

    while(chico > ze) {
        chico += 0.02;
        ze += 0.03;
        anos++;
    }

    printf("Serão necessários %d anos para que Zé seja maior que Chico.\n", anos);

    return 0;
}
```

5. Construir um algoritmo que calcule a média aritmética de vários valores inteiros positivos, lidos externamente. O final da leitura acontecerá quando for lido um valor negativo.

```
#include <stdio.h>

int main() {
    int valor, cont = 0, soma = 0;

    printf("Digite um valor inteiro positivo (ou um valor negativo para sair): ");
    scanf("%d", &valor);

    while(valor >= 0) {
        soma += valor;
        cont++;

        printf("Digite um valor inteiro positivo (ou um valor negativo para sair): ");
        scanf("%d", &valor);
    }

    if(cont > 0) {
        printf("A média aritmética é: %.2f\n", (float)soma / cont);
    } else {
        printf("Nenhum valor positivo foi inserido.\n");
    }

    return 0;
}
```

6. Em uma eleição presidencial existem quatro candidatos. Os votos são informados através de códigos. Os dados utilizados para a contagem dos votos obedecem à seguinte codificação: - 1,2,3,4 = voto para os respectivos candidatos; - 5 = voto nulo; - 6 = voto em branco; Elabore um algoritmo que leia o código do candidato em um voto. Calcule e escreva: - total de votos para cada candidato; - total de votos nulos; - total de votos em branco; Como finalizador do conjunto de votos, tem-se o valor 0.

```
#include <stdio.h>

int main() {
    int voto, candidatos[4] = {0, 0, 0, 0}, nulos = 0, brancos = 0;

    printf("Digite o código do voto (0 para sair): ");
    scanf("%d", &voto);

    while(voto != 0) {
        if(voto >= 1 && voto <= 4) {
            candidatos[voto-1]++;
        } else if(voto == 5) {
            nulos++;
        } else if(voto == 6) {
            brancos++;
        }

        printf("Digite o código do voto (0 para sair): ");
        scanf("%d", &voto);
    }

    printf("Total de votos para cada candidato:\n");
    for(int i = 0; i < 4; i++) {
        printf("Candidato %d: %d votos\n", i+1, candidatos[i]);
    }
    printf("Total de votos nulos: %d\n", nulos);
    printf("Total de votos em branco: %d\n", brancos);

    return 0;
}
```

7. Escreva um algoritmo que calcule a média aritmética das 3 notas dos alunos de uma classe. O algoritmo deverá ler, além das notas, o código do aluno e deverá ser encerrado quando o código for igual a zero.

```
#include <stdio.h>

int main() {
    int codigo;
    float nota1, nota2, nota3, media;

    printf("Digite o código do aluno (0 para sair): ");
    scanf("%d", &codigo);

    while(codigo != 0) {
        printf("Digite a primeira nota: ");
        scanf("%f", &nota1);
        printf("Digite a segunda nota: ");
        scanf("%f", &nota2);
        printf("Digite a terceira nota: ");
        scanf("%f", &nota3);

        media = (nota1 + nota2 + nota3) / 3;
        printf("A média do aluno %d é: %.2f\n", codigo, media);

        printf("Digite o código do aluno (0 para sair): ");
        scanf("%d", &codigo);
    }

    return 0;
}
```

8. Escreva um algoritmo que calcule a média dos números digitados pelo usuário, se eles forem pares. Termine a leitura se o usuário digitar zero (0).

```
#include <stdio.h>

int main() {
    int num, cont = 0, soma = 0;

    printf("Digite um número (0 para sair): ");
    scanf("%d", &num);

    while(num != 0) {
        if(num % 2 == 0) {
            soma += num;
            cont++;
        }

        printf("Digite um número (0 para sair): ");
        scanf("%d", &num);
    }

    if(cont > 0) {
        printf("A média dos números pares é: %.2f\n", (float)soma / cont);
    } else {
        printf("Nenhum número par foi inserido.\n");
    }

    return 0;
}
```

9. Escreva um algoritmo que leia 50 valores e encontre o maior e o menor deles. Mostre o resultado.

```
#include <stdio.h>

int main() {
    int num, maior, menor;

    printf("Digite um número: ");
    scanf("%d", &num);

    maior = num;
    menor = num;

    for(int i = 1; i < 50; i++) {
        printf("Digite um número: ");
        scanf("%d", &num);

        if(num > maior) {
            maior = num;
        }

        if(num < menor) {
            menor = num;
        }
    }

    printf("O maior número é: %d\n", maior);
    printf("O menor número é: %d\n", menor);

    return 0;
}
```

10. Escreva um algoritmo que leia o código de um aluno e suas três notas. Calcule a média ponderada do aluno, considerando que o peso para a maior nota seja 4 e para as duas restantes, 3. Mostre o código do aluno, suas três notas, a média calculada e uma mensagem "APROVADO" se a média for maior ou igual a 5 e "REPROVADO" se a média for menor que 5. Repita a operação até que o código lido seja negativo.

```
#include <stdio.h>
int main() {
    int codigo;
    float nota1, nota2, nota3, media;
    printf("Digite o código do aluno (negativo para sair): ");
    scanf("%d", &codigo);
    while(codigo >= 0) {
        printf("Digite a primeira nota: ");
        scanf("%f", &nota1);
        printf("Digite a segunda nota: ");
        scanf("%f", &nota2);
        printf("Digite a terceira nota: ");
        scanf("%f", &nota3);
        if(nota1 >= nota2 && nota1 >= nota3) {
            media = (nota1 * 4 + nota2 * 3 + nota3 * 3) / 10;
        } else if(nota2 >= nota1 && nota2 >= nota3) {
            media = (nota1 * 3 + nota2 * 4 + nota3 * 3) / 10;
        } else {
            media = (nota1 * 3 + nota2 * 3 + nota3 * 4) / 10;
        }
        printf("Código do aluno: %d\n", codigo);
        printf("Notas: %.2f, %.2f, %.2f\n", nota1, nota2, nota3);
        printf("Média: %.2f\n", media);
        if(media >= 5) {
            printf("APROVADO\n");
        } else {
            printf("REPROVADO\n");
        }
        printf("Digite o código do aluno (negativo para sair): ");
        scanf("%d", &codigo);
    }
    return 0;
}
```

11. Escreva um algoritmo que leia um número n (número de termos de uma progressão aritmética), a1 (o primeiro termo da progressão) e r (a razão da progressão) e escreva os n termos desta progressão, bem como a soma dos elementos.

```
#include <stdio.h>
int main() {
    int n, a1, r, termo, soma = 0;

    printf("Digite o número de termos: ");
    scanf("%d", &n);
    printf("Digite o primeiro termo: ");
    scanf("%d", &a1);
    printf("Digite a razão: ");
    scanf("%d", &r);

    termo = a1;

    for(int i = 0; i < n; i++) {
        printf("%d ", termo);
        soma += termo;
        termo += r;
    }

    printf("\nSoma dos elementos: %d\n", soma);

    return 0;
}
```

12. Escrever um algoritmo que leia 20 valores para uma variável  $n$  e, para cada um deles, calcule a tabuada de 1 até  $n$ . Mostre a tabuada na forma:  $1 \times n = n$   $2 \times n = 2n$   $3 \times n = 3n$  .....  $n \times n = n^2$ .

```
#include <stdio.h>

int main() {
    int n, valor;

    for(int i = 0; i < 20; i++) {
        printf("Digite um valor: ");
        scanf("%d", &valor);

        for(n = 1; n <= valor; n++) {
            printf("%d x %d = %d\n", n, valor, n * valor);
        }

        return 0;
    }
}
```

13. Escrever um algoritmo que leia um número  $n$  que indica quantos valores devem ser lidos a seguir. Para cada número lido, mostre uma tabela contendo o valor lido e o fatorial deste valor.

```
#include <stdio.h>

long long fatorial(int n) {
    long long fat = 1;
    for(int i = 2; i <= n; i++) {
        fat *= i;
    }
    return fat;
}

int main() {
    int n, valor;

    printf("Digite quantos valores devem ser lidos: ");
    scanf("%d", &n);

    for(int i = 0; i < n; i++) {
        printf("Digite um valor: ");
        scanf("%d", &valor);
        printf("O fatorial de %d é: %lld\n", valor, fatorial(valor));
    }

    return 0;
}
```

14. Escrever um algoritmo que leia um número não determinado de valores e calcule a média aritmética dos valores lidos, a quantidade de valores positivos, a quantidade de valores negativos e o percentual de valores negativos e positivos. Mostre os resultados.

```
int main() {
    int num, cont = 0, pos = 0, neg = 0, soma = 0;

    printf("Digite um número (0 para sair): ");
    scanf("%d", &num);

    while(num != 0) {
        soma += num;
        cont++;

        if(num > 0) {
            pos++;
        } else if(num < 0) {
            neg++;
        }

        printf("Digite um número (0 para sair): ");
        scanf("%d", &num);
    }

    if(cont > 0) {
        printf("Média aritmética: %.2f\n", (float)soma / cont);
        printf("Quantidade de valores positivos: %d\n", pos);
        printf("Quantidade de valores negativos: %d\n", neg);
        printf("Percentual de valores positivos: %.2f%%\n", (float)pos / cont * 100);
        printf("Percentual de valores negativos: %.2f%%\n", (float)neg / cont * 100);
    } else {
        printf("Nenhum número foi inserido.\n");
    }

    return 0;
}
```

15. Escrever um algoritmo que leia uma quantidade desconhecida de números e conte quantos deles estão nos seguintes intervalos: [0,25], [26,50], [51,75] e [76,100]. A entrada de dados deve terminar quando for lido um número negativo.

```
#include <stdio.h>

int main() {
    int num, cont1 = 0, cont2 = 0, cont3 = 0, cont4 = 0;

    printf("Digite um número (negativo para sair): ");
    scanf("%d", &num);

    while(num >= 0) {
        if(num >= 0 && num <= 25) {
            cont1++;
        } else if(num >= 26 && num <= 50) {
            cont2++;
        } else if(num >= 51 && num <= 75) {
            cont3++;
        } else if(num >= 76 && num <= 100) {
            cont4++;
        }

        printf("Digite um número (negativo para sair): ");
        scanf("%d", &num);
    }

    printf("Quantidade de números no intervalo [0,25]: %d\n", cont1);
    printf("Quantidade de números no intervalo [26,50]: %d\n", cont2);
    printf("Quantidade de números no intervalo [51,75]: %d\n", cont3);
    printf("Quantidade de números no intervalo [76,100]: %d\n", cont4);

    return 0;
}
```



16. Escrever um algoritmo que lê um conjunto não determinado de valores, um de cada vez, e escreve uma tabela com cabeçalho, que deve ser repetido a cada 20 linhas. A tabela conterá o valor lido, seu quadrado, seu cubo e sua raiz quadrada.

```
#include <stdio.h>
#include <math.h>

int main() {
    int num, cont = 0;

    printf("Digite um número (negativo para sair): ");
    scanf("%d", &num);

    while(num >= 0) {
        if(cont % 20 == 0) {
            printf("\nNúmero\tQuadrado\tCubo\tRaiz Quadrada\n");
        }

        printf("%d\t%d\t%d\t%.2f\n", num, num * num, num * num * num, sqrt(num));

        cont++;

        printf("Digite um número (negativo para sair): ");
        scanf("%d", &num);
    }

    return 0;
}
```

17. Escrever um algoritmo que lê um número não determinado de pares de valores m,n, todos inteiros e positivos, um par de cada vez, e calcula e escreve a soma dos n inteiros consecutivos a partir de m inclusive.

```
#include <stdio.h>

int main() {
    int m, n, soma;

    printf("Digite um par de valores m e n (m e n positivos, m = 0 para sair): ");
    scanf("%d %d", &m, &n);

    while(m > 0) {
        soma = 0;
        for(int i = m; i <= m + n; i++) {
            soma += i;
        }

        printf("A soma dos %d inteiros consecutivos a partir de %d é: %d\n", n, m, soma);

        printf("Digite um par de valores m e n (m e n positivos, m = 0 para sair): ");
        scanf("%d %d", &m, &n);
    }

    return 0;
}
```

18. Escrever um algoritmo que lê um número não determinado de valores para m, todos inteiros e positivos, um de cada vez. Se m for par, verificar quantos divisores possui e escrever esta informação. Se m for ímpar e menor do que 10 calcular e escrever o fatorial de m. Se m for ímpar e maior ou igual a 10 calcular e escrever a soma dos inteiros de 1 até m.

```
#include <stdio.h>
int numDivisores(int m) {
    int cont = 0;
    for(int i = 1; i <= m; i++) {
        if(m % i == 0) {
            cont++;
        }
    }
    return cont;
}
int fatorial(int m) {
    int fat = 1;
    for(int i = 2; i <= m; i++) {
        fat *= i;
    }
    return fat;
}
int somaInteiros(int m) {
    int soma = 0;
    for(int i = 1; i <= m; i++) {
        soma += i;
    }
    return soma;
}
int main() {
    int m;
    printf("Digite um número inteiro positivo (0 para sair): ");
    scanf("%d", &m);
    while(m > 0) {
        if(m % 2 == 0) {
            printf("O número %d possui %d divisores.\n", m, numDivisores(m));
        } else if(m < 10) {
            printf("O fatorial de %d é: %d\n", m, fatorial(m));
        } else {
            printf("A soma dos inteiros de 1 até %d é: %d\n", m, somaInteiros(m));
        }

        printf("Digite um número inteiro positivo (0 para sair): ");
        scanf("%d", &m);
    }
    return 0;
}
```

19. Faça um algoritmo que leia uma quantidade não determinada de números positivos. Calcule a quantidade de números pares e ímpares, a média de valores pares e a média geral dos números lidos. O número que encerrará a leitura será zero.

```
#include <stdio.h>
int main() {
    int num, cont = 0, pos = 0, neg = 0, soma = 0, somaPos = 0;
    printf("Digite um número (0 para sair): ");
    scanf("%d", &num);
    while(num != 0) {
        soma += num;
        cont++;
        if(num > 0) {
            pos++;
            somaPos += num;
        } else if(num < 0) {
            neg++;
        }
        printf("Digite um número (0 para sair): ");
        scanf("%d", &num);
    }
    if(cont > 0) {
        printf("Média geral: %.2f\n", (float)soma / cont);
        printf("Quantidade de valores positivos: %d\n", pos);
        printf("Quantidade de valores negativos: %d\n", neg);
        if(pos > 0) {
            printf("Média dos valores positivos: %.2f\n", (float)somaPos / pos);
        } else {
            printf("Nenhum valor positivo foi inserido.\n");
        }
        printf("Percentual de valores positivos: %.2f%%\n", (float)pos / cont * 100);
        printf("Percentual de valores negativos: %.2f%%\n", (float)neg / cont * 100);
    } else {
        printf("Nenhum número foi inserido.\n");
    }
    return 0;
}
```

20. Faça um algoritmo que leia vários números inteiros e calcule o somatório dos números negativos. O fim da leitura será indicado pelo número 0.

```
#include <stdio.h>

int main() {
    int num, soma = 0;

    printf("Digite um número (0 para sair): ");
    scanf("%d", &num);

    while(num != 0) {
        if(num < 0) {
            soma += num;
        }

        printf("Digite um número (0 para sair): ");
        scanf("%d", &num);
    }

    printf("A soma dos números negativos é: %d\n", soma);

    return 0;
}
```

21. Faça um algoritmo que leia vários números inteiros e positivos e calcule o produtório dos números pares. O fim da leitura será indicado pelo número 0.

```
#include <stdio.h>

int main() {
    int num, prod = 1;

    printf("Digite um número (0 para sair): ");
    scanf("%d", &num);

    while(num != 0) {
        if(num > 0 && num % 2 == 0) {
            prod *= num;
        }

        printf("Digite um número (0 para sair): ");
        scanf("%d", &num);
    }

    printf("O produtório dos números pares é: %d\n", prod);

    return 0;
}
```

22. Foi feita uma pesquisa entre os habitantes de uma região. Foram coletados os dados de idade, sexo (M/F) e salário. Faça um algoritmo que informe: a) a média de salário do grupo; b) maior e menor idade do grupo; c) quantidade de mulheres com salário até R\$100,00. Encerre a entrada de dados quando for digitada uma idade negativa. (Use o comando enquanto-faça e não use vetores ou matrizes).

```
#include <stdio.h>
int main() { int idade, menor_idade, maior_idade, cont_mulheres = 0, cont = 0; char sexo;
float salario, soma_salarios = 0;
printf("Digite a idade (negativa para sair): ");
scanf("%d", &idade);
if(idade >= 0) {
    printf("Digite o sexo (M/F): ");
    scanf(" %c", &sexo);
    printf("Digite o salário: ");
    scanf("%f", &salario);
    menor_idade = idade;
    maior_idade = idade;
    soma_salarios += salario;
    cont++;
    if(sexo == 'F' && salario <= 100) { cont_mulheres++; } }
while(idade >= 0) {
    printf("Digite a idade (negativa para sair): ");
    scanf("%d", &idade);
    if(idade >= 0) {
        printf("Digite o sexo (M/F): ");
        scanf(" %c", &sexo);
        printf("Digite o salário: ");
        scanf("%f", &salario);
        if(idade < menor_idade) { menor_idade = idade; }
        if(idade > maior_idade) { maior_idade = idade; }
        soma_salarios += salario;
        cont++;
        if(sexo == 'F' && salario <= 100) { cont_mulheres++; } } if(cont > 0) {
        printf("Média de salário do grupo: %.2f\n", soma_salarios / cont);
        printf("Menor idade do grupo: %d\n", menor_idade);
        printf("Maior idade do grupo: %d\n", maior_idade);
        printf("Quantidade de mulheres com salário até R$100,00: %d\n", cont_mulheres);
    } else {
        printf("Nenhum dado foi inserido.\n"); } return 0;
}
```

23. Foi realizada uma pesquisa de algumas características físicas da população de uma certa região, a qual coletou os seguintes dados referentes a cada habitante para serem analisados: - sexo (masculino e feminino) - cor dos olhos (azuis, verdes ou castanhos) - cor dos cabelos (louros, castanhos, pretos) - idade.

```
#include <stdio.h>
typedef struct {
    char sexo;
    char cor_olhos;
    char cor_cabelos;
    int idade;
} Habitante;

int main() {
    int n;
    printf("Digite o número de habitantes: ");
    scanf("%d", &n);

    Habitante habitantes[n];

    for(int i = 0; i < n; i++) {
        printf("Digite o sexo do habitante %d (M/F): ", i+1);
        scanf(" %c", &habitantes[i].sexo);
        printf("Digite a cor dos olhos do habitante %d (A/V/C): ", i+1);
        scanf(" %c", &habitantes[i].cor_olhos);
        printf("Digite a cor dos cabelos do habitante %d (L/C/P): ", i+1);
        scanf(" %c", &habitantes[i].cor_cabelos);
        printf("Digite a idade do habitante %d: ", i+1);
        scanf("%d", &habitantes[i].idade);
    }

    return 0;
}
```

24. Faça um algoritmo que determine e escreva: - a maior idade dos habitantes - a quantidade de indivíduos do sexo feminino cuja idade está entre 18 e 35 anos inclusive e que tenham olhos verdes e cabelos louros. O final do conjunto de habitantes é reconhecido pelo valor -1 entrada como idade.

```
#include <stdio.h>
typedef struct {
    char sexo;
    char cor_olhos;
    char cor_cabelos;
    int idade;
} Habitante;
int main() {
    Habitante habitante;
    int maior_idade = 0, cont_mulheres = 0;
    printf("Digite a idade do habitante (-1 para sair): ");
    scanf("%d", &habitante.idade);
    while(habitante.idade != -1) {
        printf("Digite o sexo do habitante (M/F): ");
        scanf(" %c", &habitante.sexo);
        printf("Digite a cor dos olhos do habitante (A/V/C): ");
        scanf(" %c", &habitante.cor_olhos);
        printf("Digite a cor dos cabelos do habitante (L/C/P): ");
        scanf(" %c", &habitante.cor_cabelos);
        if(habitante.idade > maior_idade) {
            maior_idade = habitante.idade;
        }
        if(habitante.sexo == 'F' && habitante.idade >= 18 && habitante.idade <= 35 && habitante.cor_olhos == 'V' && habitante.cor_cabelos == 'L') {
            cont_mulheres++;
        }
        printf("Digite a idade do habitante (-1 para sair): ");
        scanf("%d", &habitante.idade);
    }
    printf("A maior idade dos habitantes é: %d\n", maior_idade);
    printf("Quantidade de mulheres com idade entre 18 e 35 anos, olhos verdes e cabelos louros: %d\n", cont_mulheres);

    return 0;
}
```

25. Uma empresa deseja aumentar seus preços em 20%. Faça um algoritmo que leia o código e o preço de custo de cada produto e calcule o preço novo. Calcule também, a média dos preços com

e sem aumento. Mostre o código e o preço novo de cada produto e, no final, as médias. A entrada de dados deve terminar quando for lido um código de produto negativo. (Use o comando enquanto-faça)

```
#include <stdio.h>

int main() {
    int codigo;
    float preco, soma_sem_aumento = 0, soma_com_aumento = 0;
    int cont = 0;

    printf("Digite o código do produto (negativo para sair): ");
    scanf("%d", &codigo);

    while(codigo >= 0) {
        printf("Digite o preço de custo do produto: ");
        scanf("%f", &preco);

        soma_sem_aumento += preco;
        preco *= 1.2;
        soma_com_aumento += preco;
        cont++;

        printf("Código do produto: %d\n", codigo);
        printf("Preço novo do produto: %.2f\n", preco);

        printf("Digite o código do produto (negativo para sair): ");
        scanf("%d", &codigo);
    }

    if(cont > 0) {
        printf("Média dos preços sem aumento: %.2f\n", soma_sem_aumento / cont);
        printf("Média dos preços com aumento: %.2f\n", soma_com_aumento / cont);
    } else {
        printf("Nenhum produto foi inserido.\n");
    }

    return 0;
}
```

26. Escreva um algoritmo que gere o números de 1000 a 1999 e escreva aqueles que dividido por 11 dão resto igual a 5.

```
#include <stdio.h>

int main() {
    for(int i = 1000; i <= 1999; i++) {
        if(i % 11 == 5) {
            printf("%d\n", i);
        }
    }

    return 0;
}
```

27. Escreva um algoritmo que leia 500 valores inteiros e positivos e: a) encontre o maior valor; b) encontre o menor valor; c) calcule a média dos números lidos.

```

#include <stdio.h>

int main() {
    int num, maior, menor, soma = 0;

    printf("Digite um número inteiro positivo: ");
    scanf("%d", &num);

    maior = num;
    menor = num;

    for(int i = 1; i < 500; i++) {
        printf("Digite um número inteiro positivo: ");
        scanf("%d", &num);

        if(num > maior) {
            maior = num;
        }

        if(num < menor) {
            menor = num;
        }

        soma += num;
    }

    printf("O maior valor é: %d\n", maior);
    printf("O menor valor é: %d\n", menor);
    printf("A média dos números lidos é: %.2f\n", (float)soma / 500);

    return 0;
}

```

28. Escreva um algoritmo que lê um valor  $n$  inteiro e positivo e que calcula a seguinte soma:  $S := 1 + 1/2 + 1/3 + 1/4 + \dots + 1/n$ . O algoritmo deve escrever cada termo gerado e o valor final de  $S$ .

```

#include <stdio.h>

int main() {
    int n;
    double S = 0;

    printf("Digite um número inteiro positivo: ");
    scanf("%d", &n);

    for(int i = 1; i <= n; i++) {
        double termo = 1.0 / i;
        printf("Termo %d: %.2f\n", i, termo);
        S += termo;
    }

    printf("O valor final de S é: %.2f\n", S);

    return 0;
}

```

29. Escrever um algoritmo que calcule e mostre a média aritmética dos números lidos entre 13 e 73.

```
#include <stdio.h>

int main() {
    int inicio = 13;
    int fim = 73;
    int soma = 0;
    int contador = 0;

    for (int i = inicio; i <= fim; i++) {
        soma += i;
        contador++;
    }

    float media = (float)soma / contador;

    printf("A média aritmética dos números entre %d e %d é: %.2f\n", inicio, fim, media);

    return 0;
}
```

30. Escrever um algoritmo que gera e escreve os números ímpares entre 100 e 200.

```
#include <stdio.h>

int main() {
    for (int i = 100; i <= 200; i++) {
        if (i % 2 != 0) {
            printf("%d\n", i);
        }
    }
    return 0;
}
```

31. Escrever um algoritmo que lê 10 valores, um de cada vez, e conta quantos deles estão no intervalo [10,20] e quantos deles estão fora do intervalo, escrevendo estas informações.

```
#include <stdio.h>

int main() {
    int cont_dentro = 0;
    int cont_fora = 0;
    int valor;

    for (int i = 0; i < 10; i++) {
        printf("Digite um valor: ");
        scanf("%d", &valor);

        if (valor >= 10 && valor <= 20) {
            cont_dentro++;
        } else {
            cont_fora++;
        }
    }

    printf("Quantidade de números no intervalo [10,20]: %d\n", cont_dentro);
    printf("Quantidade de números fora do intervalo [10,20]: %d\n", cont_fora);

    return 0;
}
```

32. Escrever um algoritmo que lê 5 pares de valores a, b, todos inteiros e positivos, um par de cada vez, e com  $a < b$ , escreve os inteiros pares de a até b, incluindo o a e o b se forem pares.



```

#include <stdio.h>

int main() {
    int a, b;

    for (int i = 0; i < 5; i++) {
        printf("Digite o par de valores a e b (a < b): ");
        scanf("%d %d", &a, &b);

        printf("Números pares entre %d e %d: ", a, b);
        for (int j = a; j <= b; j++) {
            if (j % 2 == 0) {
                printf("%d ", j);
            }
        }
        printf("\n");
    }

    return 0;
}

```

33. Escrever um algoritmo que leia 20 valores para uma variável N e, para cada um deles, calcule a tabuada de 1 até N. Mostre a tabuada na forma: 1 x N = N 2 x N = 2N 3 x N = 3N ..... N x N = N<sup>2</sup>

34. Escrever um algoritmo que leia 5 conjuntos de 2 valores, o primeiro representando o número de um aluno, e o segundo representando a sua altura em centímetros. Encontre o aluno mais alto e o mais baixo. Mostre o número do aluno mais alto e do mais baixo, junto com suas alturas.

```

#include <stdio.h>

int main() {
    int N;

    for (int i = 0; i < 20; i++) {
        printf("Digite um valor para N: ");
        scanf("%d", &N);

        printf("Tabuada de 1 até %d:\n", N);
        for (int j = 1; j <= N; j++) {
            printf("%d x %d = %d\n", j, N, j * N);
        }
    }

    return 0;
}

```

35. Escrever um algoritmo que leia um conjunto de 50 informações contendo, cada uma delas, a altura e o sexo de uma pessoa (código=1, masculino código=2, feminino), calcule e mostre o

seguinte: a) a maior e a menor altura da turma b) a média da altura das mulheres c) a média da altura da turma.

```
#include <stdio.h>
int main() {
    int sexo;
    float altura, maior_altura = 0, menor_altura = 10000, soma_altura_mulheres = 0, soma_altura_turma = 0;
    int contador_mulheres = 0;

    for (int i = 0; i < 50; i++) {
        printf("Digite o sexo (1 para masculino, 2 para feminino) e a altura: ");
        scanf("%d %f", &sexo, &altura);

        if (altura > maior_altura) {
            maior_altura = altura;
        }

        if (altura < menor_altura) {
            menor_altura = altura;
        }

        if (sexo == 2) {
            soma_altura_mulheres += altura;
            contador_mulheres++;
        }

        soma_altura_turma += altura;
    }

    float media_altura_mulheres = soma_altura_mulheres / contador_mulheres;
    float media_altura_turma = soma_altura_turma / 50;

    printf("A maior altura da turma é: %.2f\n", maior_altura);
    printf("A menor altura da turma é: %.2f\n", menor_altura);
    printf("A média da altura das mulheres é: %.2f\n", media_altura_mulheres);
    printf("A média da altura da turma é: %.2f\n", media_altura_turma);

    return 0;
}
```

36. Escrever um algoritmo que leia um número N que indica quantos valores devem ser lidos a seguir. Para cada número lido, mostre uma tabela contendo o valor lido e o fatorial deste valor.

```
#include <stdio.h>
long long fatorial(int n) {
    if (n == 0)
        return 1;
    else
        return n * fatorial(n - 1);
}

int main() {
    int N, valor;

    printf("Digite o número de valores a serem lidos: ");
    scanf("%d", &N);

    for (int i = 0; i < N; i++) {
        printf("Digite um valor: ");
        scanf("%d", &valor);

        printf("Valor: %d, Fatorial: %lld\n", valor, fatorial(valor));
    }

    return 0;
}
```

37. Escrever um algoritmo que leia um valor X e calcule e mostre os 20 primeiros termos da série: 1 1 1 ..... X X<sup>2</sup> X<sup>3</sup> X<sup>4</sup>.

```

#include <stdio.h>
#include <math.h>

int main() {
    int X;

    printf("Digite um valor para X: ");
    scanf("%d", &X);

    for (int i = 1; i <= 20; i++) {
        printf("Termo %d: %.2f\n", i, pow(X, i - 1));
    }

    return 0;
}

```

38. Escrever um algoritmo que calcula e escreve o produto dos números primos entre 92 e 1478.

```

#include <stdio.h>

int eh_primo(int n) {
    if (n <= 1) return 0;
    if (n == 2) return 1;
    if (n % 2 == 0) return 0;
    for (int i = 3; i * i <= n; i += 2) {
        if (n % i == 0) return 0;
    }
    return 1;
}

int main() {
    long long produto = 1;

    for (int i = 92; i <= 1478; i++) {
        if (eh_primo(i)) {
            produto *= i;
        }
    }

    printf("O produto dos números primos entre 92 e 1478 é: %lld\n", produto);

    return 0;
}

```

39. Escrever um algoritmo que gera e escreve os 5 primeiros números perfeitos. Um número perfeito é aquele que é igual a soma dos seus divisores. (Ex.:  $6 = 1 + 2 + 3$ ;  $28 = 1 + 2 + 4 + 7 + 14$  etc).

```

#include <stdio.h>

int eh_perfeito(int n) {
    int soma = 0;
    for (int i = 1; i < n; i++) {
        if (n % i == 0) {
            soma += i;
        }
    }
    return soma == n;
}

int main() {
    int contador = 0;
    int numero = 1;

    while (contador < 5) {
        if (eh_perfeito(numero)) {
            printf("%d\n", numero);
            contador++;
        }
        numero++;
    }

    return 0;
}

```

40. Escrever um algoritmo que lê um valor n que indica quantos valores devem ser lidos para m, valores inteiros e positivos, com leitura de um valor de cada vez. Escreva uma tabela contendo o valor lido, o somatório dos inteiros de 1 até m e o fatorial de m.

```

#include <stdio.h>

int somatorio(int n) {
    int soma = 0;
    for (int i = 1; i <= n; i++) {
        soma += i;
    }
    return soma;
}

long long fatorial(int n) {
    if (n == 0)
        return 1;
    else
        return n * fatorial(n - 1);
}

int main() {
    int n, m;

    printf("Digite o número de valores a serem lidos: ");
    scanf("%d", &n);

    for (int i = 0; i < n; i++) {
        printf("Digite um valor para m: ");
        scanf("%d", &m);

        printf("Valor: %d, Somatório: %d, Fatorial: %lld\n", m, somatorio(m), fatorial(m));
    }

    return 0;
}

```

41. Faça um algoritmo que leia as três notas de 50 alunos de uma turma. Para cada aluno, calcule a média ponderada, como segue:  $MP = (n1 \cdot 2 + n2 \cdot 4 + n3 \cdot 3) / 10$ .

```

#include <stdio.h>

int main() {
    float n1, n2, n3, media_ponderada;

    for (int i = 0; i < 50; i++) {
        printf("Digite as três notas do aluno %d: ", i + 1);
        scanf("%f %f %f", &n1, &n2, &n3);

        media_ponderada = (n1 * 2 + n2 * 4 + n3 * 3) / 10;

        printf("A média ponderada do aluno %d é: %.2f\n", i + 1, media_ponderada);
    }

    return 0;
}

```

42. Faça um algoritmo que calcule a seguinte soma:  $H = 10 + 10 + 10 + \dots + 10$  O algoritmo deve ler um número  $n$  (inteiro e positivo) e mostrar o resultado final de  $H$ . A soma deve ser calculada apenas uma vez.

```

#include <stdio.h>

int main() {
    float n1, n2, n3, media_ponderada;

    for (int i = 0; i < 50; i++) {
        printf("Digite as três notas do aluno %d: ", i + 1);
        scanf("%f %f %f", &n1, &n2, &n3);

        media_ponderada = (n1 * 2 + n2 * 4 + n3 * 3) / 10;

        printf("A média ponderada do aluno %d é: %.2f\n", i + 1, media_ponderada);
    }

    return 0;
}

```

43. Fazer um algoritmo que leia 5 grupos de 4 valores (A,B,C,D) e mostre-os na ordem lida. Em seguida, ordene-os em ordem decrescente e mostre-os novamente, já ordenados.

```

#include <stdio.h>
#include <stdlib.h>
int comparar(const void * a, const void * b) {
    return ( *(int*)b - *(int*)a );
}
int main() {
    int valores[5][4];
    for (int i = 0; i < 5; i++) {
        printf("Digite 4 valores para o grupo %d: ", i + 1);
        for (int j = 0; j < 4; j++) {
            scanf("%d", &valores[i][j]);
        }
    }
    printf("\nValores na ordem lida:\n");
    for (int i = 0; i < 5; i++) {
        for (int j = 0; j < 4; j++) {
            printf("%d ", valores[i][j]);
        }
        printf("\n");
    }

    for (int i = 0; i < 5; i++) {
        qsort(valores[i], 4, sizeof(int), comparar);
    }

    printf("\nValores em ordem decrescente:\n");
    for (int i = 0; i < 5; i++) {
        for (int j = 0; j < 4; j++) {
            printf("%d ", valores[i][j]);
        }
        printf("\n");
    }

    return 0;
}

```

44. Foi feita uma estatística nas 200 principais cidades brasileiras para coletar dados sobre acidentes de trânsito. Foram obtidos os seguintes dados: - código da cidade - estado (RS, SC, PR, SP, RJ, ...) - número de veículos de passeio (em 1992) - número de acidentes de trânsito com

vítimas (em 1992) Deseja-se saber: a) qual o maior e o menor índice de acidentes de trânsito e a que cidades pertencem b) qual a média de veículos nas cidades brasileiras c) qual a média de acidentes com vítimas entre as cidades do Rio Grande do Sul.

```
#include <stdio.h>
#include <string.h>
typedef struct {
    int codigo;
    char estado[3];
    int veiculos;
    int acidentes;
} Cidade;
int main() {
    Cidade cidades[200];
    int total_veiculos = 0;
    int total_acidentes_RS = 0;
    int contador_RS = 0;
    Cidade cidade_maior_indice, cidade_menor_indice;
    for (int i = 0; i < 200; i++) {
        printf("Digite o código da cidade, estado, número de veículos e número de acidentes: ");
        scanf("%d %s %d %d", &cidades[i].codigo, cidades[i].estado, &cidades[i].veiculos, &cidades[i].acidentes);
        total_veiculos += cidades[i].veiculos;
        if (strcmp(cidades[i].estado, "RS") == 0) {
            total_acidentes_RS += cidades[i].acidentes;
            contador_RS++;
        }
        if (i == 0 || cidades[i].acidentes > cidade_maior_indice.acidentes) {
            cidade_maior_indice = cidades[i];
        }
        if (i == 0 || cidades[i].acidentes < cidade_menor_indice.acidentes) {
            cidade_menor_indice = cidades[i];
        }
    }
    float media_veiculos = (float)total_veiculos / 200;
    float media_acidentes_RS = (float)total_acidentes_RS / contador_RS;
    printf("Cidade com maior índice de acidentes: %d, com %d acidentes\n", cidade_maior_indice.codigo, cidade_maior_indice.acidentes);
    printf("Cidade com menor índice de acidentes: %d, com %d acidentes\n", cidade_menor_indice.codigo, cidade_menor_indice.acidentes);
    printf("Média de veículos nas cidades brasileiras: %.2f\n", media_veiculos);
    printf("Média de acidentes com vítimas entre as cidades do Rio Grande do Sul: %.2f\n", media_acidentes_RS);
    return 0;
}
```

45. Foi feita uma pesquisa entre os 1000 habitantes de uma região para coletar os seguintes dados: sexo (0-feminino, 1-masculino), idade e altura. Faça um algoritmo que leia as informações coletadas e mostre as seguintes informações: (use o comando repita-até) a) média da idade do

grupo; b) média da altura das mulheres; c) média da idade dos homens; d) percentual de pessoas com idade entre 18 e 35 anos (inclusive).

```
#include <stdio.h>

int main() {
    int sexo, idade, total_mulheres = 0, total_homens = 0, total_pessoas = 0, total_idade = 0, total_idade_homens = 0, total_idade_mulheres = 0, total_pessoas_18_35 = 0;
    float altura, total_altura_mulheres = 0.0;
    for (int i = 0; i < 1000; i++) {
        printf("Digite o sexo (0-feminino, 1-masculino), idade e altura: ");
        scanf("%d %d %f", &sexo, &idade, &altura);
        total_pessoas++;
        total_idade += idade;
        if (sexo == 0) {
            total_mulheres++;
            total_idade_mulheres += idade;
            total_altura_mulheres += altura;
        } else if (sexo == 1) {
            total_homens++;
            total_idade_homens += idade;
        }
        if (idade >= 18 && idade <= 35) {
            total_pessoas_18_35++;
        }
    }

    float media_idade = (float)total_idade / total_pessoas;
    float media_altura_mulheres = total_altura_mulheres / total_mulheres;
    float media_idade_homens = (float)total_idade_homens / total_homens;
    float percentual_pessoas_18_35 = (float)total_pessoas_18_35 / total_pessoas * 100;

    printf("Média da idade do grupo: %.2f\n", media_idade);
    printf("Média da altura das mulheres: %.2f\n", media_altura_mulheres);
    printf("Média da idade dos homens: %.2f\n", media_idade_homens);
    printf("Percentual de pessoas com idade entre 18 e 35 anos: %.2f%%\n", percentual_pessoas_18_35);

    return 0;
}
```

46. Foi realizada uma pesquisa de algumas características físicas da população de uma certa região. Foram entrevistadas 500 pessoas e coletados os seguintes dados: a- sexo: M (masculino) e F (feminino) b- cor dos olhos: A (azuis), V (verdes) e C (castanhos) c- cor dos cabelos: L (loiros), C (castanhos) e P (pretos) d- idade Deseja-se saber: a maior idade do grupo a quantidade de



indivíduos do sexo feminino, cuja idade está entre 18 e 35 anos e que tenham olhos verdes e cabelos louros.

```
#include <stdio.h>

typedef struct {
    char sexo;
    char cor_olhos;
    char cor_cabelos;
    int idade;
} Pessoa;

int main() {
    Pessoa pessoas[500];
    int maior_idade = 0;
    int contador = 0;

    for (int i = 0; i < 500; i++) {
        printf("Digite o sexo (M ou F), cor dos olhos (A, V ou C), cor dos cabelos (L, C ou P) e idade: ");
        scanf("%c %c %c %d", &pessoas[i].sexo, &pessoas[i].cor_olhos, &pessoas[i].cor_cabelos, &pessoas[i].idade);

        if (pessoas[i].idade > maior_idade) {
            maior_idade = pessoas[i].idade;
        }

        if (pessoas[i].sexo == 'F' && pessoas[i].idade >= 18 && pessoas[i].idade <= 35 && pessoas[i].cor_olhos == 'V' && pessoas[i].cor_cabelos == 'L') {
            contador++;
        }
    }

    printf("A maior idade do grupo é: %d\n", maior_idade);
    printf("Quantidade de indivíduos do sexo feminino, cuja idade está entre 18 e 35 anos e que tenham olhos verdes e cabelos louros: %d\n", contador);

    return 0;
}
```

47. Uma loja tem 150 clientes cadastrados e deseja mandar uma correspondência a cada um deles anunciando um bônus especial. Escreva um algoritmo que leia o nome do cliente e o valor das suas compras no ano passado e calcule um bônus de 10% se o valor das compras for menor que 500.000 e de 15 %, caso contrário.

```

#include <stdio.h>

typedef struct {
    char nome[50];
    float valor_compras;
} Cliente;

int main() {
    Cliente clientes[150];

    for (int i = 0; i < 150; i++) {
        printf("Digite o nome do cliente e o valor das suas compras no ano passado: ");
        scanf("%s %f", clientes[i].nome, &clientes[i].valor_compras);

        float bonus;
        if (clientes[i].valor_compras < 500000) {
            bonus = clientes[i].valor_compras * 0.10;
        } else {
            bonus = clientes[i].valor_compras * 0.15;
        }

        printf("O bônus do cliente %s é: %.2f\n", clientes[i].nome, bonus);
    }

    return 0;
}

```

48. Faça um algoritmo que mostre os conceitos finais dos alunos de uma classe de 75 alunos, considerando (use o comando CASO): a) os dados de cada aluno (número de matrícula e nota numérica final) serão fornecidos pelo usuário b) a tabela de conceitos segue abaixo: Nota Conceito de 0,0 a 4,9 D de 5,0 a 6,9 C de 7,0 a 8,9 B de 9,0 a 10,0 A.

```

#include <stdio.h>

typedef struct {
    int matricula;
    float nota;
} Aluno;

int main() {
    Aluno alunos[75];

    for (int i = 0; i < 75; i++) {
        printf("Digite a matrícula e a nota do aluno: ");
        scanf("%d %f", &alunos[i].matricula, &alunos[i].nota);

        char conceito;
        if (alunos[i].nota >= 9.0) {
            conceito = 'A';
        } else if (alunos[i].nota >= 7.0) {
            conceito = 'B';
        } else if (alunos[i].nota >= 5.0) {
            conceito = 'C';
        } else {
            conceito = 'D';
        }

        printf("O conceito do aluno de matrícula %d é: %c\n", alunos[i].matricula, conceito);
    }

    return 0;
}

```

49. Faça um algoritmo que lê um valor N inteiro e positivo e que calcula e escreve o fatorial de N (N!).

```

#include <stdio.h>

long long fatorial(int n) {
    if (n == 0)
        return 1;
    else
        return n * fatorial(n - 1);
}

int main() {
    int N;

    printf("Digite um valor inteiro e positivo para N: ");
    scanf("%d", &N);

    printf("O fatorial de %d é: %lld\n", N, fatorial(N));

    return 0;
}

```

50. Faça um algoritmo que leia 2 valores inteiros e positivos: X e Y. O algoritmo deve calcular e escrever a função potência  $X^Y$ .

```

#include <stdio.h>
#include <math.h>

int main() {
    int X, Y;

    printf("Digite dois valores inteiros e positivos para X e Y: ");
    scanf("%d %d", &X, &Y);

    printf("A potência  $X^Y$  é: %.2f\n", pow(X, Y));

    return 0;
}

```

51. Faça um algoritmo que calcule o fatorial de um número.

```

#include <stdio.h>

long long fatorial(int n) {
    if (n == 0)
        return 1;
    else
        return n * fatorial(n - 1);
}

int main() {
    int N;

    printf("Digite um valor inteiro e positivo para N: ");
    scanf("%d", &N);

    printf("O fatorial de %d é: %lld\n", N, fatorial(N));

    return 0;
}

```

52. Faça um algoritmo que calcule a combinação e arranjo de um conjunto de tamanho N em subconjuntos de p elementos.

```

#include <stdio.h>

long long fatorial(int n) {
    if (n == 0)
        return 1;
    else
        return n * fatorial(n - 1);
}

long long combinacao(int n, int p) {
    return fatorial(n) / (fatorial(p) * fatorial(n - p));
}

long long arranjo(int n, int p) {
    return fatorial(n) / fatorial(n - p);
}

int main() {
    int N, p;

    printf("Digite o tamanho do conjunto (N) e o número de elementos no subconjunto (p): ");
    scanf("%d %d", &N, &p);

    printf("A combinação de um conjunto de tamanho %d em subconjuntos de %d elementos é: %lld\n", N, p, combinacao(N, p));
    printf("O arranjo de um conjunto de tamanho %d em subconjuntos de %d elementos é: %lld\n", N, p, arranjo(N, p));

    return 0;
}

```

53. Faça um algoritmo que calcule os 20 primeiros números primos, dados os tres primeiros 1,2 e 3.

```

#include <stdio.h>

int eh_primo(int n) {
    if (n <= 1) return 0;
    if (n == 2) return 1;
    if (n % 2 == 0) return 0;
    for (int i = 3; i * i <= n; i += 2) {
        if (n % i == 0) return 0;
    }
    return 1;
}

int main() {
    int contador = 0;
    int numero = 1;

    printf("Os 20 primeiros números primos são:\n");
    while (contador < 20) {
        if (eh_primo(numero)) {
            printf("%d ", numero);
            contador++;
        }
        numero++;
    }

    return 0;
}

```

54. Faça um algoritmo que receba 2 números e divida o intervalo entre eles em 3 partes iguais. Obs. Faça a consistência para que os extremos não sejam iguais.

```

#include <stdio.h>

int main() {
    float num1, num2;

    printf("Digite o primeiro número: ");
    scanf("%f", &num1);

    printf("Digite o segundo número: ");
    scanf("%f", &num2);

    if (num1 != num2) {
        float ponto1 = num1 + (num2 - num1) / 3;
        float ponto2 = num1 + 2 * (num2 - num1) / 3;

        printf("Os pontos de divisão são: %.2f, %.2f\n", ponto1, ponto2);
    } else {
        printf("Os números são iguais, não é possível dividir o intervalo.\n");
    }

    return 0;
}

```

55. Faça um algoritmo que receba duas datas e retorne a diferença entre elas em dias.

```

#include <stdio.h>
typedef struct {
    int dia;
    int mes;
    int ano;
} Data;

int dias_no_mes(int mes, int ano) {
    if (mes == 4 || mes == 6 || mes == 9 || mes == 11)
        return 30;
    else if (mes == 2)
        return (ano % 400 == 0 || (ano % 100 != 0 && ano % 4 == 0)) ? 29 : 28;
    else
        return 31;
}

int diferenca_dias(Data data1, Data data2) {
    int dias = 0;
    while (data1.dia != data2.dia || data1.mes != data2.mes || data1.ano != data2.ano) {
        data1.dia++;
        dias++;
        if (data1.dia > dias_no_mes(data1.mes, data1.ano)) {
            data1.dia = 1;
            data1.mes++;
            if (data1.mes > 12) {
                data1.mes = 1;
                data1.ano++;
            }
        }
        return dias;
    }
}

int main() {
    Data data1, data2;
    printf("Digite a primeira data (dia mes ano): ");
    scanf("%d %d %d", &data1.dia, &data1.mes, &data1.ano);
    printf("Digite a segunda data (dia mes ano): ");
    scanf("%d %d %d", &data2.dia, &data2.mes, &data2.ano);
    printf("A diferença entre as duas datas é: %d dias\n", diferenca_dias(data1, data2));
    return 0;
}

```