

Universidade do Estado de Santa Catarina – UDESC
Centro de Ciências Tecnológicas CCT

Prof. André Tavares da Silva – andre.silva@udesc.br

Exercícios Resolvidos – Linguagem de Programação (LPG0001)

1) Faça um programa que leia dois números inteiros. O primeiro é o valor inicial de um contador, e o segundo é o valor final do contador (o valor inicial fornecido é inferior ao valor final). Usando o comando while, escreva na tela uma contagem que comece no primeiro número lido, escreva os números seguintes colocando apenas um número em cada nova linha da tela, até chegar ao valor final indicado.

```
#include <stdio.h>

int main() {
    int cont, vlinicial, vlfinal;
    /* final é palavra reservada e não pode ser usada como nome de variável */

    printf("Entre com o valor inicial: ");
    scanf("%f", &vlinicial);
    printf("Entre com o valor final: ");
    scanf("%f", &vlfinal);

    cont = vlinicial;
    while(cont <= vlfinal) {
        printf("%d\n", cont);
        cont++;
    }

    return 0;
}
```

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

```
#include <stdio.h>

int main() {
    int v1, v2, v3, v4, v5, negativos;
    /* ainda não vimos como fazer vetores - precisamos de 5 variáveis distintas */

    negativos = 0;
    printf("Entre com 5 valores inteiros: ");
    scanf("%d %d %d %d %d", &v1, &v2, &v3, &v4, &v5);
    if(v1<0) negativos++;
    if(v2<0) negativos++;
    if(v3<0) negativos++;
    if(v4<0) negativos++;
    if(v5<0) negativos++;

    printf("Foram informados %d números negativos", negativos);

    return 0;
}
```

Universidade do Estado de Santa Catarina – UDESC
Centro de Ciências Tecnológicas CCT

Prof. André Tavares da Silva – andre.silva@udesc.br

Exercícios Resolvidos – Linguagem de Programação (LPG0001)

3) Escrever um algoritmo que gere e escreva os números múltiplos de 11 entre 100 e 200.

```
#include <stdio.h>

int main() {
    printf("Múltiplos de 11 entre 100 e 200: ");

    for(int i=100; i<=200; i++)
        if(i%11 == 0)
            printf("%d ", i);

    return 0;
}
```

OU

```
#include <stdio.h>

int main() {
    printf("Múltiplos de 11 entre 100 e 200: ");

    for(int i=110; i<=200; i+=11)
        printf("%d ", i);

    return 0;
}
```

4) Num frigorífico existem 10 bois. Cada boi traz preso no seu pescoço um cartão contendo um número de identificação e seu peso. Implementar um algoritmo que escreva o número e o peso do boi mais gordo e do boi mais magro (não é necessário armazenar os dados de todos os bois).

```
#include <stdio.h>

int main() {
    int i, id, id_mais_gordo, id_mais_magro;
    float peso, peso_mais_gordo, peso_mais_magro;

    peso_mais_magro = 100000000.0; /* numero bem grande */
    peso_mais_gordo = -1.0 * peso_mais_magro; /* numero bem pequeno (negativo) */

    for(int i=0; i<10; i++) { /* conta de 0 a 9 = 10 vezes */
        printf("Entre com o identificador e o peso do boi: ");
        scanf("%d %f", &id, &peso);
        if(peso > peso_mais_gordo) {
            id_mais_gordo = id;
            peso_mais_gordo = peso;
        }
        if(peso < peso_mais_magro) {
            id_mais_magro = id;
            peso_mais_magro = peso;
        }
        printf("Mais magro é %d com %f Kg", id_mais_magro, peso_mais_magro);
        printf("Mais gordo é %d com %f Kg", id_mais_gordo, peso_mais_gordo);
    }
    return 0;
}
```

Universidade do Estado de Santa Catarina – UDESC
Centro de Ciências Tecnológicas CCT

Prof. André Tavares da Silva – andre.silva@udesc.br

Exercícios Resolvidos – Linguagem de Programação (LPG0001)

5) Escrever um algoritmo que escreve os números **primos** entre 100 e 200, bem como a soma destes números.

```
#include <stdio.h>

int main() {
    int soma=0, primo;
    printf("Múltiplos de 11 entre 100 e 200: ");

    for(int i=100; i<=200; i++) {
        primo = 1;
        for(int j=2; j<i; j++)
            if(i%j == 0) {
                primo = 0;
                break;
            }

        if(primo) {
            printf("%d ", i);
            soma+=i;
        }
    }
    printf("\nsoma: %d\n", soma);
    return 0;
}
```