

Lista de Exercícios – 6

Disciplina Linguagem de Programação - I

André Rossi Korol – 01810067

1)

```
#include <stdio.h>

int main()
{
    puts("Dec | Char");
    puts("-----");

    for (int i = 33; i <= 255; i++)
    {
        printf("%3d | %3c\n", i, i);
    }

    return 0;
}
```

2)

O programa mostra o alfabeto em letras minúsculas.

3)

```
#include <stdio.h>

int main()
{
    char primeiro, segundo;

    puts("Entre o primeiro caractere: ");
    scanf(" %c", &primeiro);
    puts("Entre o segundo caractere: ");
    scanf(" %c", &segundo);

    if (primeiro < segundo)
    {
        printf("Na tabela ASCII, %c vem antes de %c\n", primeiro, segundo);
    }
    else if (primeiro > segundo)
    {
        printf("Na tabela ASCII, %c vem depois de %c\n", primeiro,
segundo);
    }
    else
    {
        printf("%c eh igual a %c\n", primeiro, segundo);
    }

    return 0;
}
```

4)

```
#include <stdio.h>

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

    puts("Entre inteiros, ou entao qualquer valor nao inteiro para terminar  
seu input:");

    while (scanf("%d", &numero))
    {
        soma += numero;
    }

    printf("Somatorio = %d\n", soma);

    return 0;
}
```

5)

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

int main()
{
    unsigned int n;

    puts("Digite o numero: ");
    scanf("%u", &n);

    puts("Os primos sao:");

    // Printa a quantidade de 2s que dividem n
    while (n % 2 == 0)
    {
        printf("%d\n", 2);
        n /= 2;
    }

    // n deve ser impar neste ponto, entao podemos pular um
    // elemento (i.e., podemos fazer com que i += 2 no loop)
    for (int i = 3; i <= sqrt(n); i += 2)
    {
        // Enquanto i divide n, printa i e divide n por i
        while (n % i == 0)
        {
            printf("%d\n", i);
            n /= i;
        }
    }

    // Condicao para os casos em que n eh primo e maior que 2
    if (n > 2)
        printf("%d\n", n);

    return 0;
}
```

6)

```
#include <stdio.h>

int main()
{
    unsigned int x, y, i, resultado = 1;

    puts("Entre o valor de x:");
    scanf("%u", &x);
    puts("Entre o valor de y:");
    scanf("%u", &y);

    for (i = 0; i < y; i++)
        resultado *= x;

    printf("%u ^ %u = %u\n", x, y, resultado);

    return 0;
}
```

7)

```
#include <stdio.h>

int main()
{
    puts("Dec | Hx | Char");
    puts("-----");
    for (int i = 0; i <= 127; i++)
        printf("%3d | %2x | %c\n", i, i, i);

    return 0;
}
```

8)

```
#include <stdio.h>

int main()
{
    int n, x, pares = 0;
    puts("Entre a quantidade de numeros a ser entrada:");
    scanf("%d", &n);

    for (int i = 0; i < n; i++)
    {
        printf("Entre o %d° numero:\n", i + 1);
        scanf("%d", &x);
        if (!(x & 1))
            pares++;
    }

    printf("Quantidade de numeros pares: %d\n", pares);

    return 0;
}
```

Algumas das razoes para nao ter usado a expressao `continue`:

[ESLint no-continue](#)

[Continue in nested while loops](#)

[Why is continue inside a loop a bad idea?](#)

9)

```
#include <stdio.h>

int main()
{
    int n, i, x, maior, menor;

    puts("Entre o valor (n) de numeros a serem entrados:");
    scanf("%d", &n);

    for (i = 0; i < n; i++)
    {
        printf("Entre o %dº numero:\n", i + 1);
        scanf("%d", &x);

        if (i == 0)
        {
            maior = x;
            menor = x;
        }
        else
        {
            if (x > maior)
                maior = x;
            if (x < menor)
                menor = x;
        }
    }

    printf("Maior = %d\nMenor = %d\n", maior, menor);

    return 0;
}
```

10)

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

void handleFirstIteration(int *x, int *maior, int *menor)
{
    *maior = *x;
    *menor = *x;
}

void handleMaiorMenor(int *x, int *maior, int *menor)
{
    *maior = (*x > *maior) ? *x : *maior;
    *menor = (*x < *menor) ? *x : *menor;
}

int main()
{
    int n, i;
    int *x = (int *)malloc(sizeof(int));
    int *maior = (int *)malloc(sizeof(int));
    int *menor = (int *)malloc(sizeof(int));

    puts("Entre o valor (n) de numeros a serem entrados:");
    scanf("%d", &n);

    for (i = 0; i < n; i++)
    {
        printf("Entre o %d° numero:\n", i + 1);
        scanf("%d", x);
        (i == 0) ? handleFirstIteration(x, maior, menor) :
handleMaiorMenor(x, maior, menor);
    }

    printf("Maior = %d\nMenor = %d\n", *maior, *menor);

    free(x);
    free(maior);
    free(menor);

    return 0;
}
```


11)

a)

na	ca	i	s	c	n	m
3	2	1	0	0	8	----
3	2	1	8	1	7	----
3	2	1	15	2	7	----
3	2	1	15	2	7	7.50
3	2	2	0	0	10	7.50
3	2	2	10	1	9	7.50
3	2	2	19	2	9	7.50
3	2	2	19	2	9	9.50
3	2	3	0	0	4	9.50
3	2	3	4	1	5	9.50
3	2	3	9	2	5	9.50
3	2	3	9	2	5	4.50

b)

O problema que o fluxograma estaria resolvendo é o problema de calcular as médias de um dado número de alunos, para um dado número de notas.

c) & d)

```
#include <stdio.h>

int main()
{
    int na, ca, i, c;
    float m, s, n;

    puts("Entre o numero de alunos:");
    scanf("%d", &na);

    puts("Entre o numero de notas a serem entradas para cada aluno:");
    scanf("%d", &ca);

    for (i = 1; i <= na; i++)
    {
        s = 0;
        c = 0;

        while (c < ca)
        {
            printf("Entre a nota de numero %d do %d° aluno:\n", c + 1, i);
            scanf("%f", &n);
            s += n;
            c++;
        }

        m = s / c;

        printf("Media do %d° aluno: %.2f\n\n", i, m);
    }

    return 0;
}
```