

Segunda Lista

1. Qual é a diferença entre um *loop* (comando repetitivo) com condição de entrada e um *loop* com condição de saída?

2. O que o seguinte fragmento de código imprimiria se fosse parte de um programa válido?

```
int i;
for (i = 0; i < 5; i++)
    printf("%d\n", i);
```

3. O que o seguinte fragmento de código imprimiria se fosse parte de um programa válido?

```
int j;
for (j = 0; j < 11; j += 3)
    printf("%d", j);
printf("\n%d\n", j);
```

4. O que o seguinte fragmento de código imprimiria se fosse parte de um programa válido?

```
int j = 5;
while ( ++j < 9)
    printf("%d\n", j++);
```

5. O que o seguinte fragmento de código imprimiria se fosse parte de um programa válido?

```
int k = 8;
do
    printf(" k = %d\n", k);
while (k++ < 5);
```

6. Escreva um *loop* for que imprima os valores 1 2 4 8 16 32 64 incrementando o valor de uma variável contadora por um fator de 2 cada ciclo.

7. Como você faz o corpo de um *loop* incluir mais de um comando?

8. Escrever um programa em C para ler a altura, o sexo ('m', 'M', 'f' ou 'F') e a idade de 20 pessoas. Imprimir, para cada pessoa, sua altura se ela for do sexo masculino e a sua idade se for do sexo feminino. Na hipótese de ser digitado um valor diferente de 'f', 'F', 'm', 'M', o programa deve imprimir uma mensagem de erro explicativa dizendo "código do sexo está errado." Use o comando **switch**.

9. *Número perfeito* é aquele cuja soma de seus divisores, exceto ele próprio, resulta no número. Por exemplo, o 6 é um número perfeito pois seus divisores são 1, 2, 3 e 6. A soma deles, exceto o 6, resulta em 6. Sabe-se que a cada intervalo 10^n até 10^{n+1} , $n \geq 0$, existe apenas um número perfeito, ou seja, de 1 a 10 existe apenas um (que é o 6), de 10 a 100 também, etc. Escreva um programa a partir de um algoritmo otimizado, que forneça os 4 primeiros números perfeitos.

ICMC-USP
4a. Lista de Exercícios
SCC-120 (continuação)

10. Escrever um programa que lê o texto a partir do teclado, imprimindo apenas os caracteres diferentes de branco. Assuma o caractere ponto como final de texto.
11. Escreva um programa para contar o número de entradas das teclas A, B ou C, maiúsculas ou minúsculas, durante sua execução. O programa termina quando se entrar com a tecla ESC cujo código ASCII é 27.
12. Faça um programa na linguagem C que peça, repetidamente, que o usuário entre com dois números inteiros e escreva a mensagem “Soma 13” sempre que a soma destes dois números seja 13. Use como condição de parada soma maior que 100.
13. Faça um programa na linguagem C que, a cada caractere entrado, verifique se o mesmo é uma letra maiúscula, uma letra minúscula ou um dígito. Conte e imprima o número de vezes de cada ocorrência. O programa deve contar também o número de ocorrências de caracteres diferentes de letras e números. Use como condição de parada a entrada do caractere '.' (ponto).
14. Faça um programa na linguagem C que calcule a soma (SOMA) de 10 números inteiros fornecidos pelo usuário. A seguir, escreva os números entre 50 e 300 que sejam divisores de SOMA.
15. Faça um programa na linguagem C que calcule o número médio de horas extras semanais cumpridas pelos operários de uma empresa. Para cada operário leia o número de horas trabalhadas na semana. Considera-se como extras todas as horas trabalhadas acima de 40. Como condição de parada tem-se número de horas trabalhadas = -1.
16. Considere o seguinte fragmento de código:

```
int linha = 0;
char ch;
while ((ch = getchar()) != EOF)
{
    if (ch == 'Q')
        break;
    if (ch != '\n')
        continue;
    linha++;
}
```

Rescreva este código sem usar `break` ou `continue`.