

Lista de Exercício

Leitura de Código

Questão 01 (1xp) - qual a saída do código abaixo?

```
#include <stdio.h>

void funcao( int a, int b )
{
    a = 2;
    b = 1;
}

int main( void )
{
    int a = 1, b = 2;
    funcao( a, b );
    printf( "%i, %i", a, b );
    return 0;
}
```

Questão 02 (1xp) - qual a saída do código abaixo?

```
#include <stdio.h>

int funcao( int a )
{
    int b = a + 10;
    int c = b * 5;
    a = c;
    return a;
}

1.
int main( void )
{
    int a = 10;
    int b = a;
    int c = funcao( b );
    printf( "%i %i %i", a, b, c );
    return 0;
}
```

Questão 03 (1xp) - qual a saída do código abaixo?

```
#include <stdio.h>

void imprime_linha( int n )
{
    int i;
    for( i = 0; i < n; ++i )
        printf( "-" );
    printf( "\n" );
}
```

```

}

int main( void )
{
    imprime_linha( 1 );
    imprime_linha( 3 );
    imprime_linha( 5 );
    imprime_linha( 7 );
    imprime_linha( 5 );
    imprime_linha( 3 );
    imprime_linha( 1 );
    return 0;
}

```

Questão 04 (2xp) - qual a saída produzida pelo código abaixo?

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

```

int main(){
    int x = 10;
    int y = 6;
    int z = x+y/2;

    for(y=1; y<x; y++){
        if(z < x){
            break;
        }
        else{
            z--;
        }
    }

    printf("%d %d %d", x, y, z);

    return 0;
}

```

Questão 05 (1xp) - qual a saída produzida pelo código abaixo?

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

```

int main(){
    int x = 1;
    int y = 1;
    int z;

    z = x+y > 2 ? ++y : x++;

    printf("%d\n", z);

    return 0;
}

```

Questão 06 (2xp) - qual a saída do código abaixo?

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

```
char proxima_letra(char letra){  
    printf("%c", letra);
```

```
    switch(letra){  
        case 'A':  
            return 'D';  
        case 'D':  
            return 'O';  
        case 'E':  
            return 'N';  
        case 'H':  
            return 'A';  
        case 'K':  
            return 'E';  
        case 'O':  
            return 'U';  
        case 'U':  
            return 'K';  
        case 'N':  
            return '\n';  
    }
```

```
}
```

```
int main(){  
    char a = 'H';
```

```
    a = proxima_letra(a);  
    a = proxima_letra(a);  
    a = proxima_letra(a);  
    a = proxima_letra(a);  
    a = proxima_letra(a);  
    a = proxima_letra(a);  
    a = proxima_letra(a);  
    a = proxima_letra(a);
```

```
    printf("%c", a);
```

```
    return 0;
```

```
}
```

Ache o erro

Questão 01 (1xp)

A função abaixo deve retornar o quadrado de 10. Qual o problema dela e como ele pode ser corrigido?

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

```

int quadrado( int v )
{
    v = v * v;
}

int main( void )
{
    printf( "%i", quadrado( 10 ) );
    return 0;
}

```

Questão 02 (1xp)

Qual o erro de sintaxe do código abaixo?

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

```

int soma( int a, b, c )
{
    return a + b + c;
}

int main( void )
{
    int x = 15;
    printf( "%i", soma( 10, 5, x ) );
    return 0;
}

```

Questão 03 (2xp)

O programa abaixo deveria receber um valor inteiro e imprimir se o número é primo ou não.

Qual o erro com o código abaixo?

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

```

int eh_primo(int n){
    int i;

    for(i=2; i<n; i++){
        if(n%i == 0){
            return 0;
        }
    }

    return 1;
}

int main(){
    int n, ok;

    scanf("%d", &n);

    ok = eh_primo(n);
}

```

```

    if(ok == 0){
        printf("%d eh primo\n", n);
    }
    else{
        printf("%d nao eh primo\n", n);
    }

    return 0;
}

```

Complete o código!

Questão 01 (1xp)

Implemente a função **mais_alto**, para fazer com que o programa abaixo imprima o valor da maior altura digitada pelo usuário.

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

função mais_alto

```

int main(){
    int n, i;
    float altura, maior;

    scanf("%d", n);
    maior = 0.00;

    for(i=0; i<n; i++){
        scanf("%f", &altura);
        maior = mais_alto(altura, maior);
    }

    printf("%.2f\n", maior)

    return 0;;
}

```

Questão 02 (2xp)

Complete o código de forma que o programa abaixo imprima um número ao contrário. Por exemplo, se o valor digitado for 251, ele imprimirá 152, se for 100 ele imprimirá 001.

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

assinatura da função{

```

    int digito;

    while(num > 0){
        digito = num%10;
        printf("%d", digito);
        comando1;
    }

```

```

}

int main(){
    int num;

    comando2;

    reverte_numero(num);

    return 0;
}

```

Problemas para resolver

Questão 01 - Kodesh (1xp)

Escreva um programa que lê um número inteiro não negativo **n** e chama função que calcule o fatorial de **n**. Lembrando que:

$$n! = 1*2*3*4*...*n$$

Entrada:

0
1
2
3
4

Saída:

1
1
2
6
24

Questão 02 (2xp)

Escreva um programa que:

- lê o peso (em kg) e a altura (em metros) de uma pessoa
- calcula o índice de massa corporal (IMC) desta pessoa
- imprime o IMC e a classificação desta pessoa segundo os critérios abaixo

OBS:

- O IMC de uma pessoa é calculado dividindo-se o peso (em kg) pela altura ao quadrado (em metros)
- A classificação segundo o IMC é a seguinte:
 - Magreza quando o IMC está abaixo de 19
 - Peso ideal quando o IMC está entre 20 e 25
 - Acima do peso quando o IMC está entre 26 e 30
 - Obesidade quando o IMC ultrapassa 31
- Definir função que recebe o peso e a altura como parâmetros e retorna o IMC
- Definir função que recebe o IMC como parâmetro e imprime a classificação

Questão 03 (2xp)

O professor Girão está atrás de um aluno que possa ajudá-lo em um projeto para o laboratório de embarcados: ele construiu um robô para participar de uma corrida em um labirinto! Devido às tarefas da coordenação, ele ainda não teve tempo de programar os

comandos de movimentação do robô, e pediu que a turma de ITP fizesse isso para ele!

O robô irá aceitar três comandos de movimentação:

M - ele move para frente

D - ele vira para a direita

E - ele vira para a esquerda

Qualquer comando que não for um desses três, deverá ser ignorado!

O robô sempre começa com a direção para o norte. As direções podem ser mapeadas da seguinte forma:

N - 0

L - 1

S - 2

O - 3

Considere que cada comando de movimento M anda uma unidade, e que os comandos D e E apenas alteram a direção do robô, sem mudar a posição.

Construa um programa que leia uma posição inicial (x,y) do robô e uma sequência de 10 instruções de movimentação. Ao final, imprima a posição x,y que o robô terminou.

Entrada:

0 0

MMMMMMMMMM

0 0

DMMMMMMMMM

0 0

DDMMMMMMMM

0 0

EMMMMMMMMM

Saída:

0 10

9 0

0 -8

-9 0

Questão 04 - URI 1131 (adaptado) (2xp)

O campeonato potiguar de futebol é conhecido por suas fortes emoções! As partidas são extremamente disputadas, e de vez em quando eles conseguem até marcar um gol! Curioso com o desempenho das equipes locais, o presidente da Federação Potiguar de Futebol pediu que você escreva um programa para mostrar as estatísticas de resultado entre os principais times da capital!

Seu programa deverá ler o resultado de várias partidas entre ABC e América, e informar ao final:

- Quantas partidas o América venceu
- Quantas partidas o ABC venceu
- Quantas partidas acabaram em empate
- Quem foi o grande vencedor dos confrontos, ou informar que ninguém venceu mais que o outro.

Seu programa irá ler um valor inteiro n indicando quantas partidas serão lidas. Após isso, serão informadas n linhas contendo dois valores inteiros: o número de gols marcados pelo América e o número de gols marcados pelo ABC.

Entrada:

3
0 2
1 0
1 1

1
2 3

4
0 2
3 0
1 0
2 2

Saída

América venceu: 1
ABC venceu: 1
Empates: 1
Ninguém é o rei dos clássicos

América venceu: 0
ABC venceu: 1
Empates: 0
ABC é o rei dos clássicos

América venceu: 2
ABC venceu: 1
Empates: 1
América é o rei dos clássicos

Questão 05 (3xp)

Issac está bastante chateado com o seu marceneiro! Já é a terceira vez que ele encomenda o armário e o mesmo é construído com as medidas erradas: Ou as gavetas ficam muito folgadas, ou não cabem no armário! Sendo ele mesmo um grande carpinteiro, Isaac decidiu ele mesmo montar o seu armário.

Para isso, ele encomendou todas as peças do seu armário a uma fábrica de peças de madeira. Isaac mandou as dimensões da porta e das gavetas e está esperando as peças ficarem prontas. Enquanto sua encomenda não chega, ele decidiu escrever um programa que pudesse verificar se as gavetas foram feitas corretamente ou não. Como a época de provas está chegando, ele decidiu deixar essa tarefa como exercício para a turma de ITP!

Seu programa deverá ler um valor inteiro n , que representa o número de gavetas que Isaac encomendou. Então, serão entrados $n+1$ valores: as dimensões do seção de gavetas do armário (altura e comprimento) seguido pelas dimensões das n gavetas (altura e comprimento).

O programa deverá verificar se todas as gavetas cabem ou não no armário!

- Caso todas as gavetas caibam, seu programa deverá imprimir: **Gavetas OK**
- Caso exista uma folga de espaço, seu programa deve imprimir: **Gavetas Folgadas**
- Caso as gavetas não caibam no armário, seu programa deve imprimir: **Gavetas Não Cabem**

Entrada

4
10 5
2 5
3 5

Saída

Gavetas OK

3 5
2 5

3
5 3
1 3
3 3
1 5

Gavetas Não Cabem

5
20 6
3 6
4 6
1 6
6 6
3 6

Gavetas Folgadas

2
3 4
2 3
1 4

Gavetas Folgadas