

1) Reescreva a função abaixo, de forma que ela produza *exatamente o mesmo resultado*, mas sem utilizar o operador ternário.

```
int gera_codigo (int x, int y) {  
    return (x >= y) ? 10*x + y : y*y;  
}
```

2) Qual o valor de retorno da função abaixo quando ela é chamada com parâmetro de entrada n=8?

```
int sequencia (int n) {  
    int i, total = 0;  
    for (i = 1; i <= n; i+=2) {  
        total += i;  
        if (i == 5) {  
            continue;  
        }  
        total *= 2;  
    }  
    return total;  
}
```

3) Escreva a função `sortear()`, que retorna 1, 2 ou 3 com probabilidade uniforme.

4) Escreva uma função que recebe como parâmetro de entrada um número inteiro, e que *não retorna nada*, mas *imprime na tela* “pedra”, se o parâmetro de entrada for 1, “papel”, se o parâmetro de entrada for 2, ou “tesoura”, se o parâmetro de entrada for 3. Se o parâmetro de entrada for qualquer outro número, a função não imprime nada.

5) Escreva um programa que simula a disputa do jogo pedra-papel-tesoura. Tesoura ganha de papel; papel ganha de pedra; e pedra ganha de tesoura. Há dois jogadores, Jogador A e Jogador B. Para cada um, o programa deve sortear um número entre 1 e 3, usando a função do exercício 3. Depois, o programa deve escrever na tela o resultado obtido por cada jogador, “traduzindo” o número no respectivo objeto, utilizando a função do exercício 4. Finalmente, o programa deve dizer que jogador ganhou. Em caso de empate, o programa deve imprimir “Empate.”.

Por exemplo: se para o Jogador 1 o número sorteado foi 2, e para o Jogador 2 o número sorteado foi 3, o programa deve escrever na tela:

```
Jogador 1 --> papel  
Jogador 2 --> tesoura  
O Jogador 2 venceu.
```

6) O seguinte programa foi feito para encontrar o maior divisor comum entre dois números digitados pelo usuário. No entanto, existem alguns erros nesse programa. Identifique esses erros e corrija-os. (Obs.: basta dizer qual é cada erro na folha de resposta, não é preciso reescrever todo o programa.)

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

int eh_divisor(int numero, int candidato_divisor) {
    if (numero % candidato_divisor == 0) {
        return 1;
    }
    return 0;
}

int menor(a, b) {
    return (a <= b) ? a : b;
}

int main() {
    int x, y;
    printf("\nDigite o primeiro inteiro: ");
    scanf("%d", &x);
    printf("\nDigite o segundo inteiro: ");
    scanf("%d", &y);

    i = menor(x, y);
    while (i > 1) {
        if (eh_divisor(x, i) & eh_divisor(y, i)) {
            break;
        }
        i--;
    }

    printf("\nMDC de %d e %d = %d", x, y, i)
}
```

7) Em muitos sites da Internet, o usuário precisa escolher uma senha para se registrar. Escreva um programa que realize o seguinte fluxo de ações:

- 1- pede ao usuário para digitar uma senha numerica
- 2- lê a senha digitada como um número inteiro;
- 3- verifica se a senha é um número entre 100000 e 999999;
- 4- se a senha não passar na verificação acima, exibe uma mensagem de erro e volta ao passo 1;
- 5- pede ao usuário que repita a senha digitada para efeito de confirmação;
- 6- verifica se a senha digitada pela segunda vez corresponde à mesma senha digitada antes;
- 7- caso as senhas digitadas não “casem”, exibe uma mensagem de erro e volta ao passo 1;
- 8- finalmente, quando uma senha válida tiver sido digitada e confirmada, imprime uma mensagem dizendo que a senha foi cadastrada com sucesso.