



1ª PROVA DE PROGRAMAÇÃO I – INF 110 (30 pontos)

NOME _____ MAT _____

1. (5 pontos) Considere o código-fonte a seguir e responda as questões que seguem.

```
#include<iostream>
using namespace std;

void troca(int a, int b) {
    a = b;
    b = a;
}

void ordena(int a, int b, int c) {
    if(a >= b) troca(a, b);
    if(b >= c) troca(b, c);
    if(a >= b) troca(a, b);
}

int main() {
    int a, b, c;
    cin >> a >> b >> c;
    ordena(a, b, c);
    cout << a << b << c;
    return 0;
}
```

a) Qual a saída do programa acima quando o usuário fornece os números 3 2 1 como entrada?

321

b) Quais modificações devem ser feitas no código para que a saída mostre sempre os números fornecidos como entrada em **ordem decrescente**? Não é necessário reescrever o código, mas apenas indicar as modificações que devem ser feitas.

1. Trocar os operadores `>=` da função `ordena` por operadores `<=` ou `<`
2. Fazer com que os parâmetros das funções `ordena` e `troca` sejam passados por referência.
3. Alterar o código da função `troca` por:

```
int aux = a;
a = b;
b = aux;
```

2. (6 pontos) O máximo divisor comum (MDC) de dois números A e B é o maior valor inteiro que divide tanto A quanto B . Escreva **uma função** em C ou C++ que recebe como entrada dois inteiros A e B e retorna o MDC de A e B .

Escreva aqui a sua solução:

```
int mdc(int a, int b) {  
    while(b != 0) {  
        int c = b;  
        b = a % b;  
        a = c;  
    }  
    return a;  
}
```

Outras soluções são possíveis para essa questão; o algoritmo acima é o de Euclides, mostrado em sala de aula.

3. (6 pontos) Escreva um programa em C ou C++ que receba como entrada dois inteiros A e B denotando um intervalo $[A, B]$, onde $A < B$. Em seguida, o programa receberá também como entrada uma sequência $A, A+1, \dots, B-1, B$. O seu programa deve imprimir a mensagem *SIM* caso a sequência esteja completa, e *NAO* caso contrário. A sequência não estará completa se faltar algum inteiro entre A e B (os valores de A e B estarão sempre presentes na sequência). Veja alguns exemplos de entrada e saída abaixo.

Entrada	Saída
20 30 20 21 22 23 24 25 26 27 28 29 30	SIM

Entrada	Saída
20 30 20 21 22 23 24 25 26 27 28 30	NAO

Entrada	Saída
3 9 3 9	NAO

Escreva aqui a sua solução:

```
int main() {
    int a, b;
    cin >> a >> b;

    int v;
    bool completa = true;
    for(int i = a; i <= b; i++) {
        cin >> v;
        if(i != v) {
            completa = false;
            break;
        }
    }

    if(completa)
        cout << "SIM" << endl;
    else
        cout << "NAO" << endl;

    return 0;
}
```

4. (7 pontos) Crie um programa em C ou C++ que recebe como entrada um inteiro $N > 0$ e imprime na tela uma matriz quadrada de tamanho $N \times N$. A matriz deverá ser impressa com o caracter '.' em todas as posições, com exceção das posições nas diagonais da matriz, onde o programa deverá utilizar o caracter 'x'. Veja alguns exemplos abaixo.

Entrada	Saída
10	x.....x .x.....x. ..x.....x.. ...x...x...xx.....xx..... ...x...x... ..x.....x.. .x.....x.. x.....x

Entrada	Saída
5	x...x .x.x. ..x.. .x.x. x...x

Entrada	Saída
2	xx xx

Escreva aqui a sua solução:

```
int main() {  
    int n; cin >> n;  
  
    for(int i = 0; i < n; i++) {  
        for(int j = 0; j < n; j++) {  
            if(i == j || i + j == n - 1)  
                cout << "x";  
            else  
                cout << ".";  
        }  
        cout << endl;  
    }  
}
```

5. (6 pontos) Considere um número de quatro dígitos $abcd$ e a seguinte propriedade:

$$abcd = (ab + cd) * (ab + cd)$$

Por exemplo, o número 2025 satisfaz essa propriedade, pois $2025 = (20 + 25) * (20 + 25)$.
Complete o código abaixo para que o programa imprima todos os números entre 1000 e 9999 que satisfaçam a propriedade acima.

```
#include <iostream>
using namespace std;

//escreva aqui o protótipo da sua função
bool propriedade(int i);

int main() {
    for(int i = 1000; i <= 9999; i++)
        if(propriedade(i))
            cout << i << endl;
}

//escreva aqui a sua função
bool propriedade(int i) {
    int soma = i/100 + i%100;
    if(soma * soma == i)
        return true;
    return false;
}
```