

Lista de exercícios 2

1. Em Java, para que serve o comando default da estrutura switch? Ele é opcional ou obrigatório? Este comando deve estar inserido em qual lugar dentro desta estrutura?
2. Explique a diferença entre o comando break e o comando continue. Existe alguma forma na linguagem Java para interromper um laço externo que esteja aninhado a um laço interno?
3. Construa um programa na linguagem Java que leia da entrada padrão um número natural e retorne na saída padrão uma mensagem indicando se o número lido é ou não um número primo.
4. Escreva em Java um programa que calcule a n-ésima posição da sequência de Fibonacci. Use uma estrutura de repetição ao invés de usar recursão. Exemplo da sequência de Fibonacci: 0, 1, 1, 2, 3, 5, 8, 13
5. Usando o comando for, faça um algoritmo que conte o número de 1's que aparecem em uma string, por exemplo: 0011001 => 3.
6. Escreva um programa em Java que leia da entrada padrão dois números inteiro A e B, respectivamente, e retorne na saída padrão a multiplicação de A por B. O programa deve utilizar apenas as operações aritméticas de incremento e soma.
7. Escreva um programa em Java que leia da entrada padrão dois números inteiro A e B, respectivamente, e retorne na saída padrão o quociente e o resto da divisão de A por B. O programa deve utilizar apenas as operações aritméticas de incremento, decremento, soma e subtração.
8. Escreva um programa em Java que repetidamente leia um número de 1 a 9 fornecido pelo usuário e calcule a tabuada do mesmo, até que o usuário digite -1 para finalizar.
9. Crie uma classe *Fahrenheit2Celsius* que tenha um método que converte uma temperatura em Fahrenheit e retorne o valor correspondente em Celsius. Crie um método main e dentro dele crie uma instância da classe. Além disso, leia uma temperatura Fahrenheit do usuário e imprima o valor em Celsius correspondente usando o método criado.

10. Qual é a resposta após a execução do código abaixo:

```
class Fors{
    static int i;
    public static void main(String args[]){
        for(i=0;i<3;i++){
            System.out.print(i);
        }
        for(int i=2; i<3;i++){
            System.out.print(i);
        }
        int i;
        for(i=0;i<2;i++){
            System.out.print(i);
        }
        System.out.print(Fors.i);
    }
}
```

11. O que acontece ao compilar e executar o programa a seguir?

```
class Fundamentos{
    public static void main(String args[]){
        int i;
        while(i<10){
            i++;
            System.out.println(i);
        }
    }
}
```

12. Observe o código abaixo:

```
for (int i = 0; i < 3; i++) {
    for (int j = 1; j < 3; j++) {
        if (i % j == 0) {
            continue;
        } else {
            System.out.println("i = " + i + ", j = " + j);
        }
    }
}
```

O que acontece ao compilar e executar o programa acima?

13. Observe o código abaixo:

```
int i=1,j=1;
externo: for (i = 1; i <= 5; i++) {
    for (j = 1; j <= 5; j++) {
        if (i+j==5) {
            break externo;
        }
    }
}
System.out.println(i+j);
```

Qual será o resultado apresentado na tela?

14. Qual será o resultado do código abaixo?

```
for (int i =0; i <3; i++) {  
    switch(i) {  
        case 0: break;  
        case 1: System.out.print("one ");  
        case 2: System.out.print("two ");  
        case 3: System.out.print("three ");  
    }  
}  
System.out.println("done");
```

15. Qual será o resultado do código abaixo?

```
int num=0;  
for (System.out.print("x"); num++ < 10; System.out.print("z")) {  
    System.out.print("y");  
}
```

16. Qual será o resultado do código abaixo?

```
for (int num=10; num < 9; num++) {  
    System.out.print(num);  
}
```

17. Observe o código abaixo:

```
class Fluxo{  
    public static void main(String args[]) {  
        char c = 69;  
        switch(c) {  
            case 1 : System.out.print("1");  
            default: System.out.print("x");  
            case 2 : System.out.print("2");  
            case 3 : System.out.print("3");  
            case 4 : System.out.print("4");  
        }  
    }  
}
```

Qual é o resultado apresentado na tela?

- a) Não compila
- b) Compila mas não executa
- c) x234
- d) x
- e) 1x234

18. O que acontece com o código abaixo?

```
public static void main(String args[]){  
    byte x = 3;  
    switch(x){  
        case 1:  
            System.out.println("x eh 1");  
        case 128:  
            System.out.println("x eh 128");  
    }  
}
```

- a) Compila e executa sem problemas
- b) Compila, mas não executa
- c) Não compila

19. Observe o código abaixo:

```
int a = 3;
int b = 6;
if (a > 4 && b++ > 10)
    System.out.println("aprendendo");
if (a > 4 || b++ > 10)
    System.out.println("estudando");
System.out.println(b++);
```

O que o programa deve imprimir na tela após a sua execução?

20. Defina os seguintes conceitos de programação orientada a objetos: Atributo, Classe e Objeto.

21. Em Java, quais variáveis são inicializadas automaticamente e quais necessitam ser inicializadas pelo desenvolvedor? Dê um exemplo de cada um e diga se eles já possuem um valor definido previamente pela linguagem.

22. Sobre modificadores de acesso:

- a) Diga quais são os modificadores de acesso possíveis para as classes e explique cada um deles.
- b) O que acontece se usarmos a palavra reservada final antes do nome de uma classe?

23. Descreva o que significa cada linha identificada do código abaixo na linguagem Java.

```
1.  class Author {
2.      static final String PUBLISHER = "Terra Nova";
      String name;
      String email;
      char gender;
5.
      Author(String name, String email, char gender) {
6.          this.name = name;
7.          this.email = email;
8.          this.gender = gender;
9.      }

      public static void main(String[] args) {
10.         Author a=new Author("Joao", "joao@abc.com", 'M');
11.
12.         System.out.println(a.name);
13.         System.out.println(a.email);
14.         System.out.println(a.gender);
15.
16.         Book b = new Book(a, "meu livro");
17.         System.out.println(b.name);
18.         System.out.println(b.author.name);
19.     }
20. }

    class Book {
        Author author;
        String name;
        Book(Author author, String name) {
            this.author = author;
            this.name = name;
        }
    }
```

24. Partindo da classe *Rectangle*:

```
class Rectangle {
    int width = 0;
    int height = 0;
    Point origin;

    // construtores
    Rectangle() {
        origin = new Point(0, 0);
    }
    Rectangle(Point p) {
        origin = p;
    }
    Rectangle(int w, int h) {
        this(new Point(0, 0), w, h);
    }
    Rectangle(Point p, int w, int h){
        origin = p;
        width = w;
        height = h;
    }

    // move o retângulo de posição
    void move(int x, int y) {
        origin.x = x;
        origin.y = y;
    }

    // retorna a área do retângulo
    int area() {
        return width * height;
    }
}
```

Dado o código a seguir, diga o que está errado. Se existir algum erro, corrija o código.

```
public static void main(String[] args) {
    Rectangle myRect;
    myRect.width = 40;
    myRect.height = 50;
    System.out.println("myRect's area is " + myRect.area());
}
```

25. Crie um código de uma função main que cria uma instância da classe abaixo, inicializa suas duas variáveis de instância, e em seguida, exibe o valor de cada variável.

```
class NumberHolder {
    int anInt;
    float aFloat;
}
```

26. O programa abaixo contém 11 erros. Encontre e corrija todos.

```
class Tricky
    public static main(String args) {
        System.out.println>Hello world);
        system.out.Pritnln("Do you like this program?");
        System.out.println()

        System.println("I wrote it myself.");
    }
}
```

27. Crie uma Classe chamada *ContaCorrente* que possua:
- a) Uma variável (instância ou classe?) que indique a porcentagem dos juros para o cheque especial
 - b) Uma variável (instância ou classe?) que indique o saldo
 - c) Um método *deposito*
 - d) Um método *saque*
 - e) Um método *debitarJurosChequeEspecial*
 - f) Um método para exibir o saldo
 - g) Crie um método *main*, crie uma instância da classe *ContaCorrente* e faça depósitos, saques e debite os juros do cheque especial.

Depois adicionar:

- h) Uma variável para o primeiro nome do cliente;
- i) Uma variável para o sobrenome do cliente;
- j) Criar método que retorna o nome do cliente completo;
- k) Uma variável para o número da conta corrente (long);
- l) Criar método que retorna o número da conta do cliente;
- m) Uma variável correspondente ao *limiteChequeEspecial* - vai alterar o método de sacar;

28. Crie uma classe chamada *Fatura* que contenha:

- a) Uma descrição do item comprado
- b) Uma quantidade de itens comprados
- c) O preço de cada item
- d) Método *getPrecoTotal* que retorna o preço total da venda (preço de cada item X quantidade)
- e) Criar um método *main* que cria duas faturas com as compras a seguir e imprima o valor total da compra:
 - e.1. 135 notebooks custando R\$1250,33 cada;
 - e.2. 375 teclados custando R\$55,50 cada;

29. De acordo com a classe a seguir:

```
class GradeBook {
    String courseName;

    GradeBook(String name){
        courseName = name;
    }

    void setCourseName(String name) {
        courseName = name;
    }

    String getCourseName() {
        return courseName;
    }

    void displayMessage() {
        System.out.printf("Curso %s!\n", getCourseName());
    }
}
```

Modifique a classe como segue:

- a) Inclua uma nova variável de instância *String* que represente o nome do instrutor do curso;

- b) Forneça um método *set* para alterar o nome do instrutor e um método *get* para recuperá-lo;
 - c) Modifique o construtor que especifique dois parâmetros (um para o nome do curso e outro para o nome do instrutor);
 - d) Modifique o método *displayMessage* de tal maneira que ele primeiro gere a saída da mensagem de boas-vindas e o nome do curso e depois a saída: "Este curso é apresentado pelo instrutor:" seguido pelo nome do instrutor;
 - e) Crie um método *main* que utilize as novas capacidades da classe.
30. Implemente um código que realize as seguintes operações usando um vetor unidimensional:
- a) Declare, construa e inicialize um vetor de inteiros com 10 posições preenchidas com 0;
 - b) Crie um vetor com 15 posições chamado *bonus* onde cada posição contém o valor do seu índice;
 - c) Crie um vetor com 5 elementos múltiplos de 10 e mostre os elementos no console.
31. Implemente um programa que leia um vetor de inteiros do console e imprima a média, a moda (caso exista apenas uma) e a mediana dos valores digitados.
32. Desenvolva um algoritmo que imprima a soma de todos os elementos em uma matriz de inteiros $n \times n$.