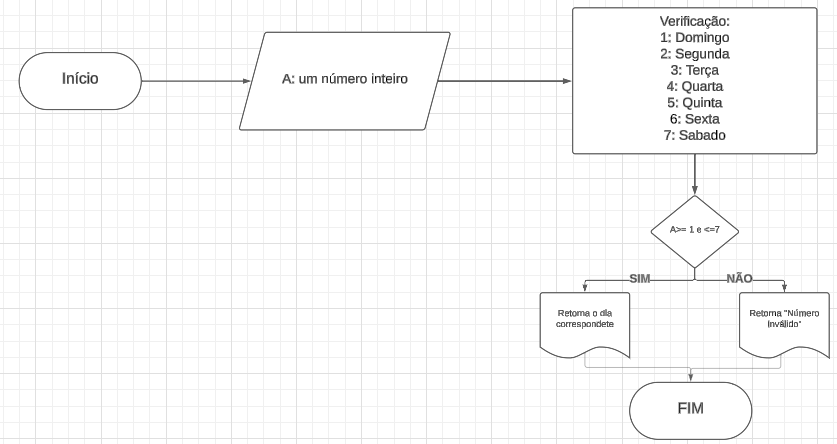
Mateus Marques Soares  
  
Simulado de prova LP1

Resolução em java: https://github.com/MateusMSoares/Linguagem-Programacao-FATEC/tree/main/Simulado/src

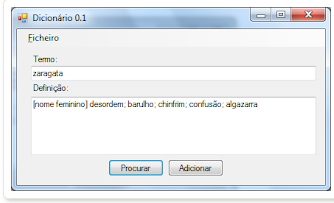
1. Construa um algoritmo em fluxograma que peça um valor compreendido entre 1 e 7 ao usuário e escreva na tela o dia da semana correspondente:

  
  
1: "Domingo"  
2: "Segunda"  
3: "Terça"  
4: "Quarta"  
5: "Quinta"  
6: "Sexta"  
7: "Sábado"  
Outro: "Inválido"

Em seguida, implemente o algoritmo em Java.

Fonte: <https://sites.google.com/site/csharpgpsi/modulo-1/exercicios>

2-Considere a tela a seguir.



<https://sites.google.com/site/csharpgpsi/modulo-7>

1. Quais componentes são utilizados nesta interface gráfica?

TextField de entrada,

TextField de saída,

Botão para procurar,

Botão para adicionar.

1. Sabendo que a tela visa coletar dados para armazenamento de atributos de um objeto, definir a classe Java deste objeto, contendo atributos, métodos construtor e de acesso.

public class Dicionario {

    private String termo;

    private String origemTermo;

    private String[] sinonimos;

    private List<Dicionario> dicionario;

    // Construtor de adicionar

    public Dicionario(String termo, String origemTermo, String[] sinonimos) {

        this.termo = termo;

        this.origemTermo = origemTermo;

        this.sinonimos = sinonimos;

    }

    // Metodo de buscar

    public Dicionario buscarInformacoesPorTermo(String termo){

        for (Dicionario dicionario : dicionario) {

            if (dicionario.termo.equals(termo)) {

                return dicionario;

            }

        }

        return null;

    }

    public String getTermo() {

        return termo;

    }

    public void setTermo(String termo) {

        this.termo = termo;

    }

    public String getOrigemTermo() {

        return origemTermo;

    }

    public void setOrigemTermo(String origemTermo) {

        this.origemTermo = origemTermo;

    }

    public String[] getSinonimos() {

        return sinonimos;

    }

    public void setSinonimos(String[] sinonimos) {

        this.sinonimos = sinonimos;

    }

    public List<Dicionario> getDicionario() {

        return dicionario;

    }

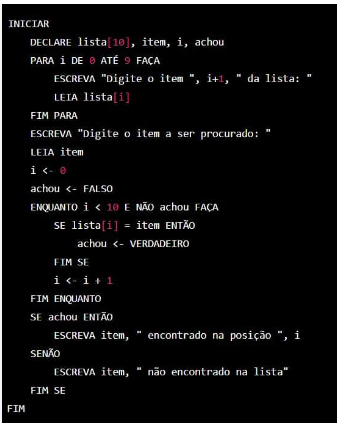
    public void setDicionario(List<Dicionario> dicionario) {

        this.dicionario = dicionario;

    }

}

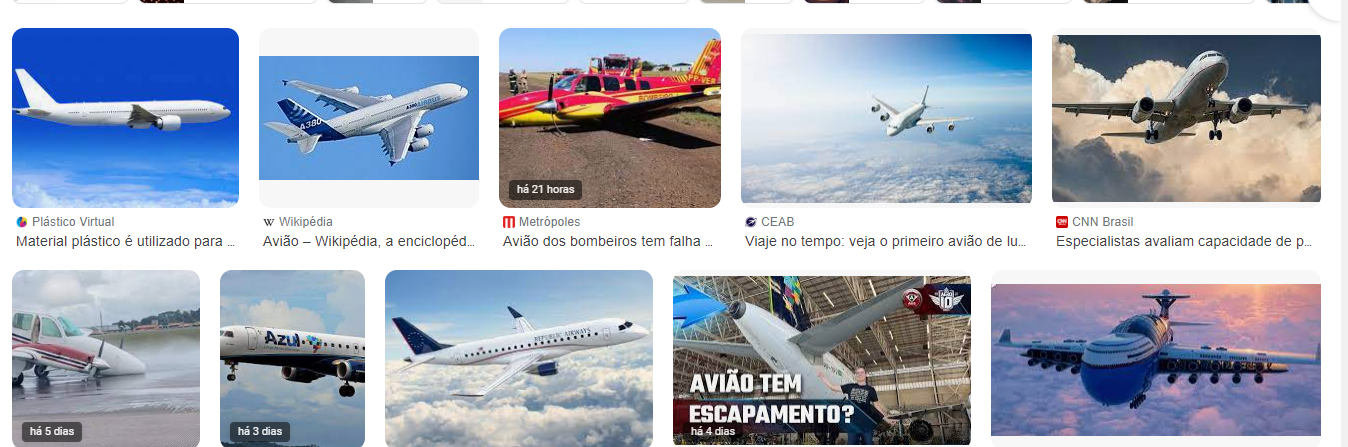
3. Faça o teste de mesa para o algoritmo a seguir.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Passo** | **i** | **lista[i]** | **item** | **achou** | **Ação** |
| 1 | - | - | - | - | Declaração de variáveis |
| 2 | 0 | 5 | - | - | Lê lista[0] = 5 |
| 3 | 1 | 8 | - | - | Lê lista[1] = 8 |
| 4 | 2 | 3 | - | - | Lê lista[2] = 3 |
| 5 | 3 | 9 | - | - | Lê lista[3] = 9 |
| 6 | 4 | 2 | - | - | Lê lista[4] = 2 |
| 7 | 5 | 7 | - | - | Lê lista[5] = 7 |
| 8 | 6 | 4 | - | - | Lê lista[6] = 4 |
| 9 | 7 | 1 | - | - | Lê lista[7] = 1 |
| 10 | 8 | 6 | - | - | Lê lista[8] = 6 |
| 11 | 9 | 0 | - | - | Lê lista[9] = 0 |
| 12 | - | - | 7 | - | Lê item = 7 |
| 13 | 0 | 5 | 7 | FALSO | Compara lista[0] com item |
| 14 | 1 | 8 | 7 | FALSO | Compara lista[1] com item |
| 15 | 2 | 3 | 7 | FALSO | Compara lista[2] com item |
| 16 | 3 | 9 | 7 | FALSO | Compara lista[3] com item |
| 17 | 4 | 2 | 7 | FALSO | Compara lista[4] com item |
| 18 | 5 | 7 | 7 | VERDADEIRO | Compara lista[5] com item e encontra |
| 19 | - | - | 7 | VERDADEIRO | Escreve "7 encontrado na posição 5" |

Em seguida, implemente em Java o algoritmo.

4 - Implemente uma classe Java que descreva os objetos a seguir.



A classe deve ter:

1. métodos construtor e de acesso
2. mínimo de três atributos
3. três métodos de classe

Instanciar três objetos dessa classe no método main.