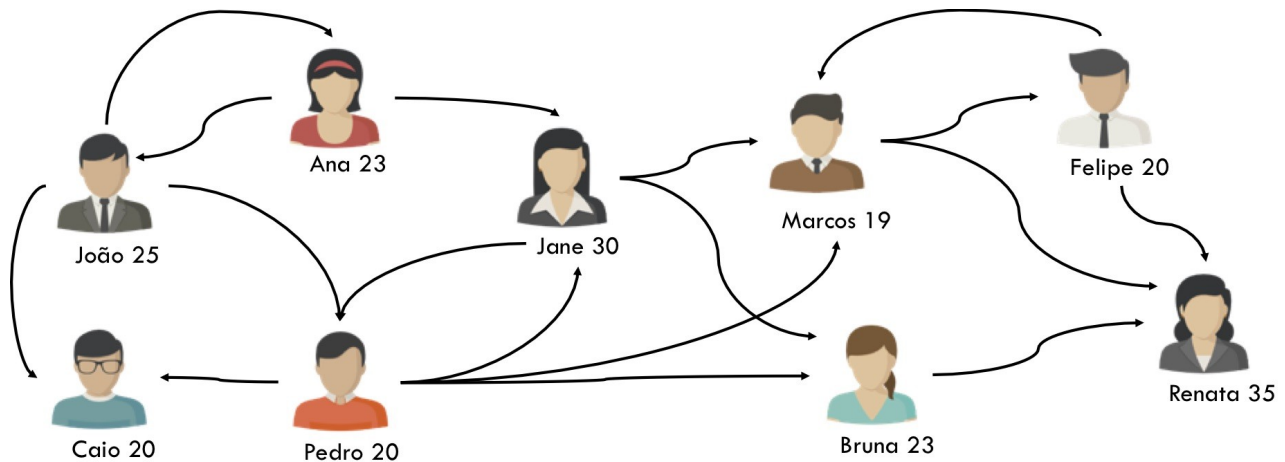


**EXERCÍCIO (ENTREGAR OS ARQUIVOS EM MÍDIA, PARA FINS DE TESTE, JUNTAMENTE COM A AD IMPRESSA):**

Considere o grafo a seguir, que representa seguidores no Instagram. Cada pessoa tem **nome** e **idade** (**nome** é o id do vértice). Uma aresta (**v1, v2**) significa que **v1** segue **v2** no Instagram. O parâmetro de entrada do seu programa é um arquivo com os nomes dos participantes do Instagram e quem cada um dos participantes segue. Para o grafo que segue:



O arquivo de entrada é representado por:

Ana 23  
Bruna 23  
Caio 20  
Felipe 20  
Jane 30  
João 25  
Marcos 19  
Pedro 20  
Renata 35  
Ana Jane  
Ana João  
Bruna Renata  
Felipe Marcos  
Felipe Renata  
Jane Bruna  
Jane Marcos  
Jane Pedro

João Ana  
João Caio  
João Pedro  
Marcos Felipe  
Marcos Renata  
Pedro Bruna  
Pedro Caio  
Pedro Jane  
Pedro Marcos

Responder às seguintes questões:

1. Quantas pessoas uma determinada pessoa segue? **public static int numero\_seguidores (TGrafo g, String nome);**
2. Quem são os seguidores de uma determinada pessoa? (função imprime os nomes dos seguidores): **public static void seguidores (TGrafo g, String nome);**
3. Quem é a pessoa mais popular? (tem mais seguidores) **public static String mais\_popular (TGrafo g);**
4. Quais são as pessoas que só seguem pessoas mais velhas do que ela própria? (função imprime os nomes das pessoas): **public static void segue\_mais\_velho(TGrafo g);**

**LEMBRE-SE: SEU PROGRAMA SÓ PODE LER O ARQUIVO DE ENTRADA UMA ÚNICA VEZ, E ELE DEVE EXECUTAR COM QUAISQUER DADOS INFORMADOS COMO PARÂMETROS DE ENTRADA. SE O SEU PROGRAMA RESOLVER SOMENTE O EXEMPLO SUPRACITADO, SUA QUESTÃO SERÁ TOTALMENTE DESCONTADA.**

**RESPOSTA:**

```
import java.io.*;
import java.util.*;

class Vizinho{
    String nome_viz;
    Vizinho prox;

    Vizinho(String nome){
        nome_viz = nome;
        prox = null;
    }

    public String toString(){ return nome_viz + "\n"; }
}

class Lista{
    String nome;
    int idade;
    Vizinho prox_viz;
    Lista prox_no;
}
```

```

Lista(String n, int id){
    nome = n;
    idade = id;
    prox_viz = null;
    prox_no = null;
}

Vizinho pertence(String nome){
    Vizinho resp = prox_viz;
    while((resp != null) && (!nome.equals(resp.nome_viz)))
        resp = resp.prox;
    return resp;
}

void ins_Viz(String n){
    Vizinho v = pertence(n);
    if(v != null) return;
    v = new Vizinho(n);
    v.prox = prox_viz;
    prox_viz = v;
}

public String toString(){
    String resp = nome + " (" + idade + "): \n";
    Vizinho p = prox_viz;
    while(p != null){
        resp += p.toString();
        p = p.prox;
    }
    return resp + "\n";
}
}

class TGrafo{
    Lista prim;

    TGrafo(){ prim = null; }

    Lista pertence(String nome){
        Lista resp = prim;
        while((resp != null) && (!nome.equals(resp.nome)))
            resp = resp.prox_no;
        return resp;
    }

    int getIdade(String nome){
        Lista resp = pertence(nome);
        if(resp == null) return -1;
        return resp.idade;
    }

    void insere(String n1, String n2){
        Lista p = pertence(n1);
        p.ins_Viz(n2);
    }
}

```

```

void insere(String n, int id){
    Lista p = pertence(n);
    if(p == null){
        p = new Lista(n, id);
        Lista q = prim;
        if(q == null){
            prim = p;
            return;
        }
        while(q.prox_no != null) q = q.prox_no;
        q.prox_no = p;
    }
}

public String toString(){
    String resp = "";
    Lista p = prim;
    while(p != null){
        resp += p.toString();
        p = p.prox_no;
    }
    return resp;
}

}

public class AD2_POO_2018_2{
    public static void main(String[] args) throws IOException{
        BufferedReader in;
        in = new BufferedReader(new FileReader(args[0]));
        try {
            TGrafo g = new TGrafo();
            String s, vs[];
            s = in.readLine();
            while(s != null){
                vs = s.split(" ");
                if((vs[1].charAt(0) == '0') || (vs[1].charAt(0) == '1') ||
(vs[1].charAt(0) == '2') || (vs[1].charAt(0) == '3') || (vs[1].charAt(0)
== '4') || (vs[1].charAt(0) == '5') || (vs[1].charAt(0) == '6') ||
(vs[1].charAt(0) == '7') || (vs[1].charAt(0) == '8') || (vs[1].charAt(0)
== '9'))
                    g.insere(vs[0], Integer.parseInt(vs[1]));
                else
                    g.insere(vs[0], vs[1]);
                s = in.readLine();
            }
            in.close();
            System.out.print(g);
            //Q1
            System.out.println("Q1: ");
            Lista l = g.prim;
            while(l != null){
                System.out.println(l.nome + " -> " + numero_seguidores (g,
l.nome));
                l = l.prox_no;
            }
        }
    }
}

```

```

        //Q2
        System.out. println("\nQ2: ");
        l = g.prim;
        while(l != null){
            System.out. print(l.nome + " -> ");
            seguidores (g, l.nome);
            l = l.prox_no;
        }
        //Q3
        System.out. println("\nQ3: ");
        System.out. println("Pessoa(s) mais popular(es): " +
mais_popular(g));
        //Q4
        System.out. println("\nQ4: ");
        segue_mais_velho(g);
    }
    catch (Exception e){
        System.out.println("Excecao\n");
    }
}

//Q1
public static int numero_seguidores (TGrafo g, String nome){
    if(g == null) return 0;
    Lista l = g.pertence(nome);
    if(l == null) return 0;
    Vizinho v = l.prox_viz;
    int resp = 0;
    while(v != null){
        resp++;
        v = v.prox;
    }
    return resp;
}

//Q2
public static void seguidores (TGrafo g, String nome){
    Lista l = g.prim;
    String resp = "";
    while(l != null){
        Vizinho v = l.pertence(nome);
        if(v != null) resp = resp + l.nome + " ";
        l = l.prox_no;
    }
    if(!resp.equals("")) System.out.println(resp);
}

//Q3
public static String mais_popular (TGrafo g){
    Lista l = g.prim;
    String mais_popular = "";
    int cont, max = -1;
    while(l != null){
        cont = 0;
        Lista laux = g.prim;

```

```

        while(laux != null){
            Vizinho v = laux.pertence(l.nome);
            if(v != null) cont++;
            laux = laux.prox_no;
        }
        if(cont > max){
            mais_popular = l.nome;
            max = cont;
        }
        else if(cont == max) mais_popular = mais_popular + ", " + l.nome;
        l = l.prox_no;
    }
    return mais_popular;
}

//Q4
public static void segue_mais_velho(TGrafo g){
    Lista l = g.prim;
    System.out. print("Pessoa(s) que só segue(m) mais velho(s): ");
    String resp = "";
    while(l != null){
        Vizinho v = l.prox_viz;
        while(v != null){
            if(g.getIdade(v.nome_viz) <= l.idade) break;
            v = v.prox;
        }
        if((v == null)&&(l.prox_viz != null)) resp = resp + l.nome + " ";
        l = l.prox_no;
    }
    if(!resp.equals("")) System.out.println(resp);
    else System.out.println("ninguém!");
}
}

```