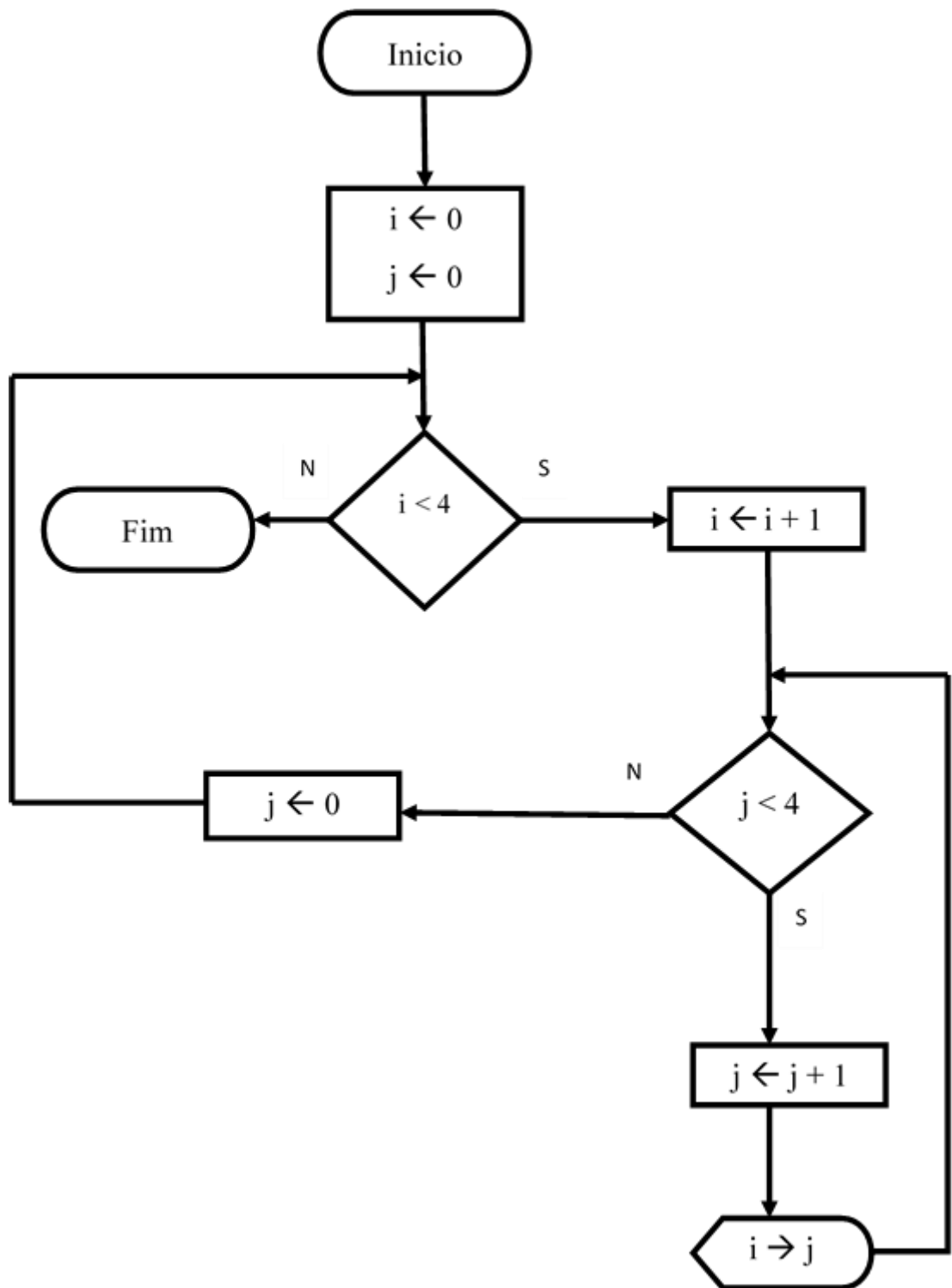


1. Escreva uma classe (com método *main*) para representar uma equação do segundo grau. A classe deverá ter o método para calcular e retornar as raízes da equação. Realize simulações. Para calcular raiz quadrada utilize o método estático `Math.sqrt()` e, para calcular potenciação o método estático `Math.pow(a, b)`. A classe `Math` está no pacote `java.lang` que é auto importado.
2. Crie um programa para verificar a situação de um aluno. Elabore uma classe `Aluno` com os seguintes atributos: nome do aluno, nota da prova 1, nota da prova 2, nota da NAC e nota da AM. A classe deverá ter os seguintes métodos: calcular e retornar a média ponderada (Provas peso 10, NAC e AM peso 5); verificar se um aluno foi aprovado (média  $\geq 6.0$ ), sendo que este método deverá retornar um valor booleano; exibir nome do aluno, situação (aprovado ou reprovado) e o valor média.
3. Teste a classe `Aluno` instanciando pelo menos três objetos em uma classe de testes. Os dados de cada aluno deverão ser informados via teclado pelo usuário do sistema.
4. Forma de reutilização de software, na qual uma nova classe é criada observando membros de uma classe existente e aprimorada com capacidades específicas”. Esta descrição se refere a qual das técnicas apresentadas pelo paradigma da Programação Orientada a Objetos (POO)?
  - a) Refatoração
  - b) Herança
  - c) Hereditariedade
  - d) Polimorfismo
  - e) Reutilização de classes
5. Em relação a declaração e uso de classes, métodos e atributos no Java, assinale a alternativa correta
  - I) A utilização da cláusula *private* não permite a herança de atributos
  - II) O modificador de acesso *abstract*, quando utilizado em uma classe, proíbe a instância dela.
  - III) Com o encapsulamento só é possível manipular os atributos da classe, através de regras bem definidas implementadas nos métodos.
  - IV) As cláusulas *extends* e *implements* são utilizadas na herança e na utilização de *interfaces*
  - V) O modificador de acesso *abstract* quando implementado nos métodos força a reescrita deles.

- a) todas estão corretas
  - b) apenas alternativas I, II e III
  - c) apenas alternativas II, III e IV
  - d) apenas alternativas III, IV e V
  - e) apenas alternativas I, IV e V
6. Desenvolver uma classe chamada Bolas contendo apenas o raio ( $r$ ) como atributo. A classe deverá ter métodos para alterar o valor do raio, retornar o valor do raio, calcular a área, o perímetro e o volume da bola. O volume da esfera é dado por:  $V_e = \frac{4}{3} \cdot \pi \cdot r^3$ , o perímetro é calculado por:  $p = 2 \cdot \pi \cdot r$ , e a área por:  $A = \pi \cdot r^2$ . Teste a classe instanciando objetos e chamando os métodos. O valor de  $\pi$  pode ser obtido a partir da constante `Math.PI`, do pacote `java.lang`.
7. Considere a classe Bolas, insira um construtor e um método para adicionar e listar bolas; elabore uma classe de teste, instancie cinco objetos da classe e adicione-os a lista.
8. Crie uma classe de nome LeiOhm, contendo: três variáveis e um método que recebe os valores de Tensão( $v$ ) e Corrente( $i$ ); calcula e imprime o valor da resistência elétrica, conforme a 1ª Lei de Ohm. Fórmula  $r = \frac{v}{i}$ .
9. Considerando a classe LeiOhm acrescente outros dois métodos para calcular Tensão( $v$ ) e Corrente( $i$ ). Crie um menu de opções; em função da escolha o programa deve mostrar as opções corretas de entradas. A aplicação deve perguntar ao usuário sobre a necessidade de realizar outros cálculos ou encerrar.
10. Considere o algoritmo na figura abaixo e responda:



Quantas vezes  $i \rightarrow j$  será demonstrada?

- a) 8 vezes
- b) 16 vezes
- c) 4 vezes
- d) 32 vezes
- e) nenhuma