Universidade Federal do Espírito Santo

Prova de Seleção

Programa de Pós-Graduação em Informática (PPGI)
Processo seletivo de 2025/2

Instruções para a prova

- A prova contém **20 questões** objetivas com cinco alternativas cada. Para cada questão, **existe apenas uma alternativa correta**.
- O tempo total para a realização desta prova é de **2 horas**. Um aviso será dado 30 minutos antes do término.
- Você receberá um cartão de respostas oficial para marcar suas respostas. Preencha-o utilizando **caneta esferográfica preta ou azul**.
- Marque apenas **uma resposta por questão**. Caso haja marcações múltiplas, a questão será anulada.
- É **proibido** o uso de celulares, smartwatches ou qualquer outro dispositivo eletrônico durante a prova.
- Guarde **todos** os dispositivos **desligados** e fora do alcance (por exemplo, em sua mochila ou bolsa).
- Apenas canetas, lápis, borracha, documento de identidade e o material fornecido pela organização da prova são permitidos sobre a mesa.
- Caso tenha dúvidas durante a prova, levante a mão e aguarde a presença de um fiscal.
- Os fiscais não estão autorizados a esclarecer questões, apenas dúvidas relacionadas ao processo.
- Ao finalizar, entregue o caderno de questões e o cartão de respostas ao fiscal. Confira se preencheu corretamente todos os campos obrigatórios no cartão de resposta antes de entregá-lo. Lembre-se também de assinar a lista de entrega.
- Mantenha silêncio durante toda a prova para não atrapalhar os colegas.
- Qualquer tentativa de comunicação ou comportamento inadequado poderá resultar na desclassificação.

Boa prova!

Questão 1: Jorge, Maurício e Claudio são profissionais liberais. Um deles é arquiteto, o outro é médico e o outro é advogado. Seus escritórios estão localizados em diferentes andares de um mesmo edifício. Os nomes de suas secretárias são, não necessariamente nesta ordem, Ana, Cecília e Jane.

Sabendo-se que:

- O escritório do advogado está localizado no andar térreo;
- Jane, ao invés de casar com seu chefe como a maioria das secretárias de fotonovelas, está noiva de Claudio e almoça todos os dias com ele, na casa da futura sogra;
- Todos os dias, Ana sobe para encontrar a secretária de Maurício, e então almoçam juntas no refeitório ao lado do escritório de Maurício;
- Ontem, Jorge mandou sua secretária descer para entregar algumas gravuras ao arquiteto.

Considerando verdadeiras as informações dadas acima, podemos concluir que:

a) Cecília é secretária de Mauricio, Claudio é advogado e Jane é secretaria de Jorge;
b) Maurício é médico, Ana é secretária de Jorge e Claudio é arquiteto;
c) Jane é secretária de Mauricio, Mauricio é advogado e Cecília é secretária de Jorge;
d) Jorge é arquiteto, Ana é secretária de Maurício e Cecília é secretária de Claudio;
e) Nenhuma das alternativas anteriores.

Questão 2: Considere as premissas verdadeiras a seguir:

- Se Ana Paula joga vôlei ou Joaquim joga videogame, então Victória vai à praia
- Hoje, Victória não foi à praia
- Se hoje é sábado, então Ana Paula joga vôlei e Caio treina boxe

Considerando as premissas apresentadas, é **correto** afirmar que:

a) Hoje é sábado e Ana Paula jogou vôlei.
b) Hoje não é sábado e Joaquim não jogou videogame.
c) Ana Paula jogou vôlei ou Joaquim jogou videogame.
d) Hoje é sábado e Joaquim jogou videogame.
e) Hoje não é sábado e Ana Paula jogou vôlei.

Questão 3: Analise a seguinte proposição: "Existe pelo menos uma universidade em que todos os cursos têm, pelo menos, 100 alunos". A negação dessa proposição é logicamente equivalente à proposição:

a) Em todas as universidades existem pelo menos um curso que possui, no máximo, 99 alunos.
b) Em no máximo uma universidade existe um curso que possui, no máximo, 101 alunos.
c) Há uma universidade em que existe pelo menos um curso com, no máximo, 99 alunos.
d) Em cada universidade existe pelo menos um curso que possui, pelo menos, 100 alunos.
e) Existe nenhuma universidade em que os cursos possuam, no máximo, 100 alunos.

Questão 4: Dada a afirmação:

"Exatamente uma pessoa entre Marcos e Heide viajou".

A negação da afirmação acima é logicamente equivalente à:

a) Ambos viajaram.
b) Ambos não viajaram.
c) Marcos ou Heide não viajou.
d) Ambos viajaram ou ambos não viajaram.
e) Pelo menos um entre Marcos e Heide viajou.

Questão 5: Considere as proposições abaixo:

- **P1:** Se João estuda lógica **ou** Maria não lê livros, então Carlos vai ao cinema.
- **P2:** Se Carlos não vai ao cinema, então Ana não sai de casa.
- **P3:** Ana saiu de casa.

Com base nessas proposições, é **correto concluir que:**

a) João estudou lógica e Maria leu livros.
b) Carlos foi ao cinema, mas João não estudou lógica.
c) João não estudou lógica e Maria leu livros.
d) Maria não leu livros, portanto Ana saiu de casa.
e) Carlos não foi ao cinema, portanto João não estudou lógica.

Questão 6: Considere o seguinte trecho de código em C-like:

```
int i, j, c;
c = 1;
for (i = 1; i < n; i++) {
   for (j = 1; j <= n; j++) {
      c = c + 1;
   }
}
```

Assumindo que a instrução c = c + 1 é O(1), qual a expressão que melhor define a ordem de complexidade do trecho de código acima?

a) $O(n \log n)$
b) $O(\log n)$
c) $O(n)$
d) $O(n^2)$
e) Nenhuma das alternativas anteriores

Questão 7: Considere as seguintes afirmações sobre programação orientada a objetos:

I. O principal objetivo do encapsulamento é ocultar os detalhes de implementação de uma classe.

II. A herança é fundamental para todos os tipos de polimorfismo

III. Polimorfismo de subtipo é a capacidade de objetos de classes diferentes responderem de forma diferente a mesma mensagem.

IV. A herança permite que uma nova classe adquira atributos e métodos definidos em uma classe existente.

É correto apenas o que se afirma em:

a) I e IV
b) II e IV
c) I, II e III
d) II e III
e) I, III e IV

Questão 8: Sobre os conceitos de tipos de dados e verificação de tipos de uma linguagem de programação, considere as afirmativas a seguir.

I. Em linguagens de programação com tipagem forte, as conversões automáticas entre tipos diferentes são realizadas com frequência para facilitar a programação
II. A tipagem dinâmica permite que uma variável seja usada como diferentes tipos de dados em momentos diferentes
III. Linguagens com tipagem estática podem ser mais seguras que linguagens com tipagem dinâmica, pois erros de tipo são detectados antes da execução do programa
IV. Em linguagens com tipagem estática, o compilador pode realizar otimizações com base nos tipos conhecidos durante a compilação, enquanto linguagens com tipagem dinâmica não têm essa capacidade.

Assinale a alternativa correta.

a) Somente II e III estão corretas
b) Somente I, II e III estão corretas
c) Somente II, III e IV estão corretas
d) Somente I, II e IV estão corretas
e) Todas estão corretas

Questão 9: Estruturas de dados como listas encadeadas, pilhas e árvores binárias são muito utilizadas para resolver problemas computacionais. Sobre as características dessas estruturas de dados, avalie as afirmativas a seguir:

I) É possível definir arbitrariamente a posição de inserção de qualquer elemento em uma árvore binária de busca.
II) Em uma pilha, o último elemento a entrar é o primeiro a sair.
III) Para remover um elemento de uma lista singularmente encadeada, deve-se alterar os elementos anterior e próximo ao elemento removido.
IV) Uma lista permite que as inserções possam ser feitas em qualquer lugar (posição), mas as remoções, não.
V) Em uma lista circular com encadeamento simples, o primeiro elemento aponta para o segundo e para o último.

Assinale a alternativa que contém, de cima para baixo, a sequência correta.

a) Apenas a afirmativa II é verdadeira
b) Apenas as afirmativas III e IV são verdadeiras
c) Apenas a afirmativa I é verdadeira
d) Apenas as afirmativas I e II são verdadeiras
e) Apenas as afirmativas II e V são verdadeiras

Questão 10: Sobre a estrutura de dados árvore de busca binária, responda:

i) Um nó da árvore binária de busca pode ter qualquer número de filhos, variando de 1 a n, com $n \in \mathbb{R}$.
ii) Na árvore de busca binária, o conteúdo dos nós à esquerda de um nó *i* são sempre menores que o conteúdo de *i*, e os elementos dos nós à direita de *i* são sempre maiores que o valor do nó *i*.
iii) Árvores de busca binária são estruturas que apresentam auto-balanceamento, sempre garantindo que operações de inserção e remoção tenham comportamento O(log n).
iv) Árvores de busca binária podem se degenerar em listas encadeadas, tendo comportamento linear (O(n)) para operações como busca e inserção.

Assinale a alternativa que contém, de cima para baixo, a sequência correta.

a) Apenas a alternativa ii está correta
b) As alternativas i, ii e iv estão corretas
c) Apenas a alternativa iii está correta
d) As alternativas ii e iv estão corretas
e) Todas as alternativas estão corretas

Questão 11: Sobre Tipos Abstratos de Dados (TAD), e implementações em C, avalie as seguintes alternativas:

i) Para garantir o encapsulamento total, a estrutura deve estar definida dentro do arquivo .h referente ao TAD em questão.
ii) A manipulação (acesso e atualização) de atributos de um TAD totalmente encapsulado podem ser feitas em qualquer programa (.c), sendo este o que implementa o TAD, ou mesmo outros programas que utilizam este TAD, sem a necessidade de métodos específicos para tais operações sobre o TAD.
iii) É impossível garantir encapsulamento total para um TAD utilizando a linguagem de programação C.
iv) Acessos a atributos de TADs opacos são feitos, por programas externos ao TAD, utilizando métodos implementados dentro do próprio TAD, popularmente conhecidos como *setters* e *getters.*

Assinale a alternativa que contém, de cima para baixo, a sequência correta.

a) Apenas a afirmativa i está correta
b) Apenas a afirmativa iii está correta
c) Apenas a alternativa iv está correta
d) As alternativas i e iv estão corretas
e) Todas as alternativas estão incorretas

Questão 12: Considere o seguinte trecho de código, implementado em C, e representando um método de um Tipo Abstrato de Dados chamado tPartida

```
void obtemDadosPartida(tPartida *p, char* nomeTimeFora, char*
nomeTimeCasa, int* pontosTimeFora, int* pontosTimeCasa)
{
   strcpy(nomeTimeFora, p->nomeTimeFora);
   strcpy(nomeTimeCasa, p->nomeTimeCasa);
   *pontosTimeFora = p->pontosTimeFora;
   *pontosTimeCasa = p->pontosTimeCasa;
}
```

Assinale a alternativa correta.

a) A função não compila porque todos os parâmetros passados são ponteiros, o que não é permitido em C.
b) Os valores das variáveis atualizadas dentro da função não podem ser realizados porque somente ponteiros foram passados como parâmetro para a função.
c) As passagens de parâmetros foram todas feitas por cópia (ou valor), e, portanto, as atualizações realizadas dentro da função não são propagadas do lado de fora (onde foi chamada).
d) Os parâmetros char* e int* representam, respectivamente, strings e inteiros passados por referência que devem ter seus valores atualizados dentro da função e tais mudanças devem ser refletidas do lado de fora da função.
e) As atualizações dos valores dos parâmetros não podem ser visíveis do lado de fora da função porque nenhum valor é retornado dela.

Questão 13: Sobre o algoritmo de ordenação Heap Sort, assinale a alternativa correta.

a) O processo de garantia da propriedade heap da estrutura (heapfy()) tem comportamento quadrático, o que impacta diretamente na ordem de crescimento do algoritmo de ordenação.
b) No melhor caso, o Heap Sort ordena os dados de entrada em ordem cúbica.
c) O Heap Sort é uma variação do Bubble Sort, e por isso, tem comportamento quadrático em todos os casos.
d) O Heap Sort é uma variação do Merge Sort, e por isso, utiliza operações de merge para garantir a ordenação linear dos elementos.
e) O algoritmo Heap Sort utiliza uma fila de prioridades (heap binário), mantendo a propriedade de heap a cada iteração i para os $n - i$ elementos restantes. A cada passo, o maior (ou menor) elemento é movido para sua posição final no final (ou início) da estrutura.

Questão 14: Considerando a estrutura lista encadeada, assinale a alternativa **incorreta**.

a) A busca em listas encadeadas é sempre realizada em tempo constante (O(1))
b) A busca em listas encadeadas é realizada em tempo linear (O(n))
c) Cada elemento (ou nó) de uma lista encadeada contém um valor e uma referência (ou ponteiro) para o próximo nó da lista.
d) Listas encadeadas permitem inserções e remoções com complexidade O(1).
e) Listas encadeadas não oferecem acesso direto aos elementos por índice.

Questão 15: Considere o trecho de código em linguagem de programação C a seguir.

```
void sort(int arr[], int n) {
   for (int i = 0; i < n - 1; i++) {
      for (int j = 0; j < n - i - 1; j++) {
         if (arr[j] > arr[j + 1]) {
            int temp = arr[j];
            arr[j] = arr[j + 1];
            arr[j + 1] = temp;
         }
      }
   }
}
```

Assinale a alternativa que apresenta, corretamente, o que esse trecho de código fará ao ser executado.

a) Ordenará o vetor de entrada, de forma crescente, em ordem quadrática (Bubble Sort)
b) Ordenará o vetor de entrada, de forma decrescente, em ordem quadrática (Bubble Sort)
c) Preencherá o vetor todo com o maior elemento
d) Entrará em loop infinito
e) Reverterá a ordem dos elementos do vetor

Questão 16: Sobre o algoritmo apresentado a seguir:

```
int busca(int arr[], int tamanho, int valor) {
   for (int i = 0; i < tamanho; i++) {
      if (arr[i] == valor) {
         return i;
      }
   }
   return -1; // Retorna -1 se o valor não for encontrado
}
```

Assinale a alternativa correta sobre o funcionamento do algoritmo:

a) A função ordena o vetor de entrada em ordem crescente.
b) A função conta o número de ocorrências do valor no arr.
c) A função verifica se o valor existe no arr, retornando a primeira posição onde ele é encontrado ou -1 caso contrário.
d) A função preenche o vetor arr com o valor em todas as posições.
e) A função encontra o maior elemento no vetor arr.

Questão 17: Sobre o conceito de Programação Orientada a Objetos (POO), analise as afirmativas a seguir.

I. Encapsulamento é o princípio que visa a proteção dos dados de um objeto, restringindo o acesso direto aos seus atributos e controlando esse acesso através de métodos públicos.
II. A herança é um pilar da POO que permite que uma classe (subclasse) adquira características (atributos e comportamentos) de outra classe (superclasse), promovendo a reutilização de código e a organização hierárquica.
III. O polimorfismo permite que um objeto se comporte de diferentes maneiras dependendo do seu tipo real ou do contexto em que é utilizado, geralmente através de sobrescrita (override) ou sobrecarga (overload) de métodos.
IV. Abstração refere-se à capacidade de um objeto ter múltiplos tipos de dados para o mesmo atributo, permitindo maior flexibilidade na manipulação de informações.

Está correto o que se afirma apenas em:

a) I e III
b) b) II e IV
c) I, II e III
d) I, III e IV
e) II, III e IV

Questão 18: Analise as características de diferentes algoritmos de ordenação e assinale a alternativa CORRETA que descreve a complexidade de tempo no pior caso e a estabilidade para o algoritmo Merge Sort.

a) $O(n^2)$ e Estável
b) O(n log n) e Estável
c) O(n log n) e Instável
d) $O(n^2)$ e Instável
e) O(log n) e Estável

Questão 19: Com base no conceito de "encapsulamento" e "herança" na Programação Orientada a Objetos, assinale a alternativa **incorreta**.

a) O encapsulamento promove a modularidade e a manutenibilidade do código, uma vez que as alterações internas em uma classe não afetam diretamente o restante do sistema, desde que sua interface pública seja mantida.
b) Através da herança, uma subclasse pode estender a funcionalidade de sua superclasse, adicionando novos atributos e métodos, ou sobrescrevendo comportamentos existentes.
c) A herança sempre leva à duplicação de código, pois a subclasse precisa reimplementar todos os métodos da superclasse, mesmo aqueles que não sofrerão alterações.
d) Modificadores de acesso como private (privado) em Java ou protected (protegido) em C++ são utilizados para aplicar o encapsulamento, controlando a visibilidade de atributos e métodos.
e) O encapsulamento permite que a lógica interna de uma classe seja modificada sem que as classes que a utilizam precisem ser alteradas, desde que a assinatura dos métodos públicos permaneça a mesma.

Questão 20: Analise as afirmações a seguir sobre as propriedades de diversos algoritmos de ordenação e assinale a alternativa **correta**:

a) O Quick Sort possui complexidade de tempo O(n log n) no pior caso, é estável e geralmente é implementado como um algoritmo in-place
b) O Merge Sort é um algoritmo estável e in-place, com complexidade de tempo O(n log n) tanto no caso médio quanto no pior caso
c) O Heap Sort garante uma complexidade de tempo O(n log n) no pior caso, é in-place, mas é um algoritmo instável
d) O Insertion Sort, embora simples e in-place, apresenta complexidade de tempo O(n log n) no pior caso para listas grandes
e) O Bubble Sort é considerado eficiente para grandes volumes de dados devido à sua complexidade O(n log n) no melhor caso, e é sempre in-place