



# XIII Olimpíada Cearense de Informática

1ª FASE - 06 a 10 de Outubro de 2025

## MODALIDADE PROGRAMAÇÃO

### **Leia atentamente as instruções:**

- Não serão permitidos empréstimos de materiais, consultas e comunicação entre os candidatos, tampouco o uso de livros e apontamentos. Relógios e aparelhos eletrônicos em geral deverão ser desligados. O não cumprimento destas exigências ocasionará a exclusão do candidato deste Exame;
- Aguarde o Aplicador da Prova autorizar a abertura do Caderno de Prova. Após a autorização, confira todas as questões antes de iniciar o Exame;
- Este Caderno de Prova contém 20 (vinte) questões objetivas, cada qual com apenas 1 (uma) alternativa correta;
- Não serão permitidas perguntas ao Aplicador da Prova sobre as questões da Prova;
- A duração desta prova será de 2 (duas) horas;
- O tempo mínimo para ausentar-se definitivamente da sala é de 1 (uma) hora;
- Ao concluir a prova, permaneça em seu lugar e comunique ao Aplicador de Prova, sinalizando com uma de suas mãos;
- Aguarde autorização para devolver o Caderno de Prova.

## 2 | XIII Olimpíada Cearense de Informática

**Questão 1.** No Restaurante Universitário da UFC, os comensais podem optar por uma fruta ou por um doce, podendo também não escolher nenhum dos dois, mas é proibido pegar os dois. Um calouro da computação, meio atrapalhado, leu a placa que dizia “pegue fruta ou doce” e pegou as duas opções, já que na lógica, o “ou” permite que os dois lados da proposição sejam verdadeiros para que tudo seja verdadeiro, logo, a placa diz que é permitido pegar uma fruta e um doce. Para evitar confusões do tipo, como a placa deve ser escrita?

- a) “Se pegar doce, então não pegue fruta”.
- b) “Não pegue fruta e doce”.
- c) “Se pegar fruta, então não pegue doce”.
- d) “Pegue fruta e não pegue doce ou pegue doce e não pegue fruta ou pegue nada”.
- e) Todas as anteriores.

---

**Questão 2.** André, Bruno, Carlos e Daniel resolveram brincar de um jogo: dentro de uma caixa, seriam colocadas 8 bolas, 6 brancas e 2 pretas. Os 4 amigos, em ordem alfabética de nomes, irão sortear uma bola cada, sem devolver as bolas de volta à caixa. O primeiro a sortear uma bola preta é o ganhador. Se 4 bolas brancas forem sorteadas em seguida inicialmente, os amigos continuam sorteando até alguém sortear uma bola preta. Qual a probabilidade de André vencer o jogo?

- a)  $5/12$
- b)  $5/16$
- c)  $5/14$
- d)  $13/27$
- e)  $3/28$

---

**Questão 3.** Rafael, certo dia, estava mexendo com números binários até que surgiu uma dúvida, quantos números poderia formar tendo um número binário de 12 bits e as seguintes condições:

- O bit mais significativo tem valor igual a 1
- O número não pode ser divisível por 128
- O número não pode ser ímpar

Quantos números Rafael consegue formar?

- a) 992
- b) 1008
- c) 1024
- d) 2032
- e) 2048

### 3 | XIII Olimpíada Cearense de Informática

**Questão 4.** Durante uma missão de emergência na estação orbital OCI-7, o androide técnico Sigma-4Z foi ativado para investigar instabilidades nos módulos de fluxo magnético da nave. Essas instabilidades afetam diretamente os escudos gravitacionais e podem levar ao colapso do motor de dobra. Para diagnosticar o problema, Sigma executa um protocolo composto por duas funções:

- Uma função chamada **verificador**, usada para comparar sinais magnéticos e determinar a diferença de intensidade magnética entre dois fluxos de energia;
- E uma função chamada **ajuste**, que calibra um sinal com base na diferença verificada e retorna o novo nível de estabilidade.

Abaixo está o pseudocódigo completo utilizado por Sigma-4Z:

```
func verificador(a, b):  
    se a > b:  
        retorne a - b  
    senão:  
        retorne b - a  
  
func ajuste(sinal):  
    sinal ← sinal + verificador(sinal, 5)  
    retorne sinal
```

```
Principal:  
    sinal ← 3  
    resultado ← ajuste(sinal)  
    resultado ← ajuste(resultado)  
    escreva(resultado)
```

Com base no que foi informado, qual será a saída do programa?

- a) 5
- b) 6
- c) 7
- d) 8
- e) 9

## 4 | XIII Olimpíada Cearense de Informática

**Questão 5.** Ao codificar um programa, Otávio se deparou com um problema lógico. Se determinada expressão P fosse verdadeira, o programa teria resultado falso. Caso contrário, se a expressão Q fosse falsa, o programa teria o resultado verdadeiro. Qual das seguintes expressões lógicas poderia representar o problema lógico de Otávio?

- A) P ou Q.
- B) não-P ou Q.
- C) não-P ou não-Q.
- D) P ou não-Q.
- E) não-P e não-Q.

---

**Questão 6.** O código Gray é um sistema de código binário inventado por Frank Gray em que, de um número para o seu sucessor, apenas um bit varia. Essa codificação foi extremamente útil durante o período em que circuitos lógicos digitais se realizavam com válvulas termoiônicas e dispositivos eletromecânicos, nos quais os contadores necessitavam de muita potência e geravam ruídos altos quando muitos bits eram alterados simultaneamente. O uso do código Gray nesse caso garantiria a alteração de apenas um bit a cada sucessão na contagem, o que reduzia o ruído e a potência necessária.

A codificação de um número binário para Código Gray é feita através de um algoritmo simples:

O primeiro bit Gray será igual ao primeiro bit binário, enquanto o segundo bit Gray será o resultado do primeiro bit binário e do segundo bit binário em uma operação de “ou exclusivo” (XOR) - Porta lógica que retorna 1 se e somente se os dois bits de entrada são diferentes entre si - e isso se repete do terceiro bit em diante.

Nesse sentido, dado uma sequência de bits binários:

1º bit Gray = 1ª bit binário

nº bit Gray = nª bit binário XOR (n-1)º bit binário

Dessa forma, como seria o número binário 011010001010100 em gray?

- a) 010111001111100
- b) 011111001111110
- c) 010111001111110
- d) 011111010011110
- e) 01010111011110

## 5 | XIII Olimpíada Cearense de Informática

**Questão 7.** Você trabalha em uma fábrica de comidas enlatadas no setor de embalagem. Seu trabalho é garantir que cada lote seja padronizado e para isso existem diversas etapas. Uma dessas etapas é garantir que o peso de todas as latas estejam exatamente com o mesmo peso. Você ficou responsável por conferir um lote em que o peso está menor que o normal, porém a balança que você normalmente usa está defeituosa e você precisará usar uma balança de Roberval (balança de dois pratos) para achar qual lata está fora do padrão. Sabendo que cada lote possui 16 latas e que apenas 1 lata está fora do padrão, qual é a quantidade mínima de pesagens que você precisa fazer para garantir com certeza qual lata é a mais leve?

- a) 1
- b) 2
- c) 3
- d) 4
- e) 5

---

**Questão 8.** Os números de Fibonacci são dados por uma sequência onde o N-ésimo número da sequência é a soma dos números N-1 e N-2 da sequência. Por exemplo:

$n_1 = 0$ ;  $n_2 = 1$ ;

$n_3 = n_2 + n_1$ , logo,  $n_3 = 0 + 1 = 1$

$n_4 = n_3 + n_2$ , logo,  $n_4 = 1 + 1 = 2$

$n_5 = n_4 + n_3$ , logo,  $n_5 = 2 + 1 = 3$

E assim por diante.

João escreveu uma função para calcular o n-ésimo número de Fibonacci:

```
func fibonacci(n):  
    se n == 0:  
        retorne 0  
    senão se n == 1:  
        retorne 1  
    senão:  
        retorne fibonacci(n-1) + fibonacci(n-2)
```

Considere que João chama a função `fibonacci(7)` para testá-la. Seguindo a recursão, quantas vezes a função `fibonacci(3)` seria chamada?

- a) Uma vez.
- b) Duas vezes.
- c) Quatro vezes.
- d) Cinco vezes.
- e) Oito vezes.

## 6 | XIII Olimpíada Cearense de Informática

**Questão 9.** Abaixo, há um algoritmo em pseudocódigo que implementa uma busca binária de forma recursiva em um vetor, assuma que o vetor foi passado por referência, e o elemento a qual buscamos é denominado por “alvo”. Inicialmente, a função deve ser chamada com “início = 0” e “fim = tamanho(vetor)”. Assuma que a variável “meio” é sempre um inteiro (armazena-se apenas a parte inteira da divisão). O índice -1 deverá ser retornado caso o alvo não seja encontrado.

```
func BuscaRec(inicio, fim, vetor, alvo):  
    se inicio > fim então:  
        retorne -1  
  
    meio = (inicio + fim)/2  
  
    se vetor[meio] == alvo:  
        retorne meio  
    senão se vetor[meio] > alvo:  
        retorne BuscaRec(inicio, meio, vetor, alvo)  
    senão se vetor[meio] < alvo:  
        retorne BuscaRec(meio, fim, vetor, alvo)
```

Analise atentamente o algoritmo e assinale a alternativa correta. Assuma que o vetor tem tamanho “n” para cálculo de complexidade.

- a) O algoritmo funciona corretamente e tem complexidade  $O(n)$ .
- b) O algoritmo funciona corretamente e tem complexidade  $O(\sqrt{n})$ .
- c) O algoritmo funciona corretamente e tem complexidade  $O(\lg(n))$ .
- d) O algoritmo funciona corretamente e tem complexidade  $O(n^2)$ .
- e) O algoritmo não funciona corretamente.

---

**Questão 10.** e exatamente 3 outros alunos. A relação de amizade é uma relação mútua, então se Ana é amiga de Bruno, Bruno também é amigo de Ana. Sabendo disso, assinale a alternativa correta.

- a) Todos os alunos têm ao menos um amigo em comum.
- b) É possível que todos os alunos sejam amigos entre si.
- c) Existem dois alunos que não compartilham nenhum amigo em comum.
- d) Selecionando qualquer grupo de 3 alunos, existe pelo menos um aluno que é amigo dos 3 ao mesmo tempo.
- e) É possível que um aluno tenha mais amigos que outro aluno dentro desse grupo.

## 7 | XIII Olimpíada Cearense de Informática

### Questão 11.

```
func F(n):  
    se n == 0: retorna 0  
    senão se n == 1: retorna 1  
    retorna F(n-2) + F(n-1)
```

```
func P(n):  
    se n == 2: retorna verdadeiro  
    se n <= 1 ou n % 2 == 0: retorna falso  
    i = 3  
    enquanto (i * i) <= n:  
        se n % i == 0: retorna falso  
        i = i + 2  
    retorna verdadeiro
```

```
# PROGRAMA 1  
valor1 = 0  
valor2 = 1  
para n de 1 até 10:  
    f = F(n)  
    valor1 = valor1 + f  
    se P(f): valor2 = valor2 * f
```

Após o fim do programa 1, qual dos dois valores será mais alto?

- a) O valor 1 será mais alto que o valor 2. valor1 = 143, valor2 = 23.
- b) O valor 1 será mais alto que o valor 2. valor1 = 88, valor2 = 23.
- c) O valor 2 será mais alto que o valor 1. valor1 = 88, valor2 = 390.
- d) O valor 2 será mais alto que o valor 1. valor1 = 143, valor2 = 195.
- e) O valor 2 será mais alto que o valor 1. valor1 = 143, valor2 = 390.

## 8 | XIII Olimpíada Cearense de Informática

**Questão 12.** Analise a seguinte função que aplica uma certa organização em uma lista

```
func organiza(lista):  
    para i de 1 até tamanho(lista):  
        se lista[i] < lista[i+1]:  
            lista[i] = lista[i+1]  
            lista[i+1] = lista[i]
```

O que a função organiza faz?

- a) Ordena a lista numa ordem crescente (de menor para maior)
- b) Ordena a lista numa ordem decrescente (de maior para menor)
- c) Move o menor item da lista para o final dela
- d) Move o maior item da lista para o final dela
- e) Nenhuma das alternativas acima.

---

**Questão 13.** Qual das seguintes opções é a negação da afirmação “Se 2 divide 10, então 10 é par.”?

- a) 10 não é par e 2 divide 10.
- b) Se 2 divide 10 então 10 não é par.
- c) 2 divide 10 ou 10 não é par.
- d) 2 não divide 10 e 10 não é par.
- e) 10 não é par ou 2 divide 10.

---

**Questão 14.** Pedro é um menino muito esperto. Ele gosta de escrever mensagens secretas para Maria com palavras codificadas para que seja difícil que os outros descubram sobre o que eles estão falando. No entanto, Pedro queria escrever uma mensagem para Maria com a palavra abelha, mas não tinha certeza de como ficaria escrito usando o método de codificação deles, considerando que ele sabe que todas essas palavras abaixo estão corretamente codificadas de acordo com sua lógica, descubra-a e ajude Pedro a codificar a palavra ABELHA.

- DADO – D B E O
- LINHA – L J O H B
- TERRENO – T F S R F N P
- COMPUTADOR – C P N P V T B D O R
- ABELHA – ?

Sabendo que você descobriu a lógica, a forma correta é:

- A) B B F L I A
- B) A C E M H B
- C) A C F L I A
- D) A H L E B A
- E) B C F M I B



## 9 | XIII Olimpíada Cearense de Informática

**Questão 15.** Felipe está estudando formas diferentes de buscas em listas.

As funções a seguir retornam o índice de um elemento qualquer em uma lista qualquer.

```
func busca(lista, elemento):  
    para i de 1 até tamanho(lista)  
        se lista[i] == elemento:  
            retorna i  
    retorna -1 # 0 elemento não foi encontrado  
  
func busca2(lista, elemento, inicio, fim):  
    se inicio > fim:  
        retorna -1 # 0 elemento não foi encontrado  
    meio = (inicio + fim) // 2  
    se lista[meio] == elemento:  
        retorna meio  
    se lista[meio] < elemento:  
        retorna busca2(lista, elemento, meio + 1, fim)  
    retorna busca2(lista, elemento, inicio, meio - 1)
```

Analise as funções dadas na questão. Para buscar um elemento em uma lista qualquer com N elementos, qual função é mais rápida?

- a) A função busca2 sempre irá encontrar um elemento qualquer mais rapidamente do que a função busca1, pois diminui o número geral de comparação de elementos.
- b) A função busca2 geralmente irá encontrar elementos mais rapidamente que a função busca1, pois diminui o número geral de comparação de elementos
- c) A função busca2 sempre irá encontrar um elemento qualquer mais rapidamente do que a função busca1, pois diminui o número geral de comparação de elementos, mas depende de uma lista organizada em ordem crescente.
- d) A função busca2 geralmente irá encontrar um elemento qualquer mais rapidamente do que a função busca1, pois diminui o número geral de comparação de elementos, mas depende de uma lista organizada em ordem crescente.
- e) A função busca2 tende a encontrar elementos em tempos semelhantes à função busca1, mas como a busca2 precisa de uma lista organizada em ordem crescente, ela acaba tendo menos casos de uso.

## 10 | XIII Olimpíada Cearense de Informática

**Questão 16.** Analise o código a seguir, que opera com base em um valor de entrada  $N$  (considere  $N$  uma potência de 2):

```
contador = 0
i = 1
enquanto i <= N:
    j = 1
    enquanto j <= N:
        contador += 1
        j += i
    i *= 2
```

Qual é a ordem do número de interações em função de  $N$  ao final do código?

- a)  $O(N)$
- b)  $O(N\sqrt{N})$
- c)  $O(N^2)$
- d)  $O(N \log N)$
- e)  $O(\log N)$

---

**Questão 17.** Considere o seguinte trecho em pseudolinguagem:

```
func fazAlgo(a, b):
    enquanto b > 0:
        resto = a % b
        a = b
        b = resto
    retorna a
```

Qual o propósito principal dessa função?

- a) Determinar o mínimo múltiplo comum de  $a$  e  $b$
- b) Calcular o máximo divisor comum de  $a$  e  $b$
- c) Verificar se  $a$  e  $b$  são números primos entre si
- d) Somar todos os divisores de  $a$  e  $b$
- e) Verificar se  $a$  é divisível por  $b$

## 11 | XIII Olimpíada Cearense de Informática

**Questão 18.** O algoritmo abaixo recebe um número  $n$  do usuário e aplica 2 regras: se o número é par ele é dividido por 2, se o número é ímpar ele é multiplicado por 3 e somado 1, fazendo isso até chegar num ponto onde ele reconhece que vai começar um loop e escrevendo quantos passos até lá.

```
func numero_e_par(a):  
    se a % 2 == 0:  
        retorna verdadeiro  
    retorna falso  
  
n = leia("Digite um número maior que 0: ")  
passos = 0  
se n > 0:  
    enquanto n > 0:  
        se numero_e_par(n):  
            n = n/2  
        senao:  
            n = n*3 + 1  
        passos = passos + 1  
    escreva("Passos até o loop: " + passos)
```

No código acima, há uma falha que faz com que nunca haja uma saída por ficar num loop eterno. Marque o item que contém a correção e o número da linha de onde deve ser corrigido.

- a) se  $a \% 2 \neq 0$ ; Linha 2.
- b) se  $n > 1$ ; Linha 9.
- c) enquanto  $n > 1$ ; Linha 10.
- d)  $n = n*3$ ; Linha 14.
- e) passos = 1; Linha 8.

## 12 | XIII Olimpíada Cearense de Informática

**Questão 19.** Sobre o código em pseudolinguagem abaixo, responda:

```
resultado = 0
lista = []

para n de 1 até 10:
    dobro = n * 2
    Se dobro % 3 == 0:
        Adicione dobro à lista

para cada elemento em lista:
    resultado = resultado + elemento

se resultado > 30:
    escreva("Maior que 30")
senão:
    escreva("Menor ou igual a 30")
```

- a) A lógica está certa. O código vai somar os valores de 1 a 10 que são divisíveis por 3 e retornar se é maior que 30 ou se é menor ou igual a 30.
- b) A lógica de acrescentar o valor de elemento em resultado está errada. O valor do resultado não vai alterar.
- c) Está errado. O programa utiliza listas, que são estruturas de dados que permitem somente leitura, então não será possível acrescentar valores na lista.
- d) O programa está correto. Ele vai multiplicar por 2 os números de 1 a 10 e, se eles forem divisíveis por 3, adicioná-los à lista. Após isso, vai somar todos os elementos na lista e verificar se o resultado é maior que 30 ou não e imprimir isso na tela.
- e) A saída esperada é "Menor ou igual a 30".

### 13 | XIII Olimpíada Cearense de Informática

**Questão 20.** Considere o seguinte Algoritmo:

$A = [3, 7, 2, 9, 5, 9, 4]$

$n = \text{Tamanho}(A)$

$x = A[1]$

$y = -\infty$

para  $i$  de 2 até  $n$ :

    se  $A[i] > x$  então

$y = x$

$x = A[i]$

    Senão se  $A[i] > y$  e  $A[i] < x$  então

$y = A[i]$

escreva( $y$ )

Assinale a alternativa que indica o valor que será impresso ao final da execução do algoritmo:

- a) 5
- b) 7
- c) 9
- d) 4
- e) 3