



CONCURSO DE ADMISSÃO  
AO  
CURSO DE FORMAÇÃO  
ENGENHARIA DA COMPUTAÇÃO



CADERNO DE QUESTÕES

2021/2022

1ª QUESTÃO

Valor: 1,0

Considere uma rede de computadores de uma empresa hipotética que recebeu o endereço de rede 130.50.0.0 (classe B) para ter acesso à Internet. Para atender a demanda da empresa por endereços de redes, foi necessário dividir esse endereço em 60 subredes.

Diante do exposto, determine:

- a) a quantidade de endereços de *hosts* que terá cada subrede;
- b) a máscara de subrede que deverá ser utilizada; e
- c) a subrede à qual pertence o endereço 130.50.65.15.

2ª QUESTÃO

Valor: 1,0

Considere o alfabeto  $\Sigma = \{a, b\}$ . Seja a linguagem  $L_1 = \{a^i b^j \text{ tal que } i \leq j \leq 2i\}$ . Por exemplo:  $aabbb \in L_1$  porque  $2 \leq 3 \leq 4$ .

Determine se a linguagem  $L_1$  é livre de contexto. Prove sua resposta.

3ª QUESTÃO

Valor: 1,0

Um grupo de espões utilizava um código especial para a comunicação entre eles. Transmitiam as mensagens percorrendo uma árvore de busca binária (BST) em pós-ordem, depois de inserir na árvore, inicialmente vazia, os termos da mensagem.

Na primeira etapa, faziam um pré-processamento em que substituíam a letra A, em decimal, pelo número 10, a letra B pelo número 11, e assim sucessivamente até a letra Z, que era substituída pelo número 35.

Depois, então, inseriam os números de dois dígitos (10 a 35) na árvore BST e, em seguida, construíam o criptograma correspondente à mensagem, percorrendo a árvore em pós ordem e escrevendo os números lidos, na ordem de leitura. Finalmente, substituíam esses números pelas letras equivalentes e as transmitiam.

Qual é a sequência de letras do alfabeto transmitida para codificar a mensagem PIRANDELO?

Seja o código abaixo de uma unidade de busca binária que recebe dois parâmetros: um *array* de valores inteiros ordenados de forma crescente e um número inteiro; e retorna a posição do número no *array*, se o número se encontrar no *array*, ou  $-1$ , caso contrário.

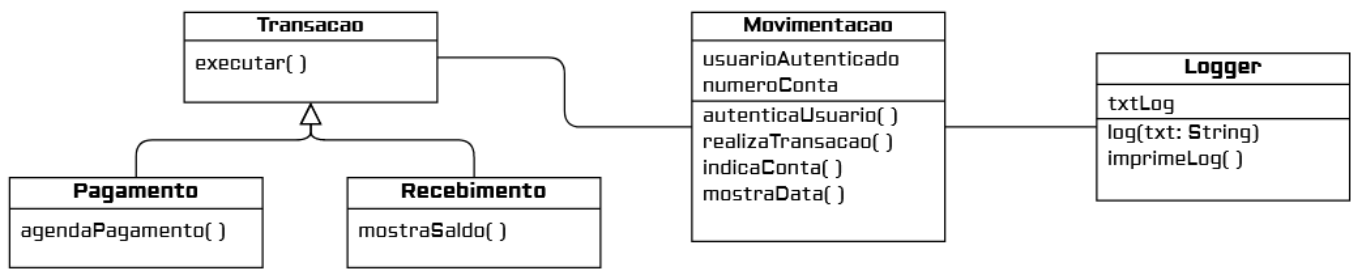
```
01. public int buscaBinaria(int[] lista, int valor) {
02.     int inicio = 0; int fim = lista.length - 1;
03.     int retorno = -1;
04.     while (inicio <= fim) {
05.         int meio = (inicio + fim) / 2;
06.         if (lista[meio] == valor) {
07.             retorno = meio;
08.             fim = -1;
09.         }
10.         else {
11.             if (valor > lista[meio]) inicio = meio + 1;
12.             else fim = meio - 1;
13.         }
14.     }
15.     return retorno;
16. }
```

**Observação:**

- Um caso de teste é um par <dados de entrada, saída esperada>.

Considerando a técnica de teste do caminho básico e a complexidade ciclomática da unidade acima, desenvolva um conjunto de casos de teste.

No conjunto de métricas CK, Chidamber e Kemerer propõem seis métricas de um projeto orientado a objetos. Considere o diagrama abaixo como sendo parte de um sistema orientado a objetos:



E o seguinte trecho da implementação da classe `Movimentacao`:

```

public class Movimentacao {
    private boolean usuarioAutenticado;
    private int numeroConta;
    ...
    public void autenticaUsuario() {
        ...
        if (numeroConta != 0) {
            usuarioAutenticado = true;
        }
    }
    public void mostraData() {
        ...
        Logger.log("guardando a data");
    }
    public void realizaTransacao() {
        ...
        if (usuarioAutenticado) {
            transacaoAtual.executar();
            ...
            Logger.imprimeLog();
        }
    }
    public Transacao indicaConta() {
        ...
        Transacao umaTr = new Recebimento(numeroConta);
        ...
    }
}
  
```

Para a classe `Movimentacao`, faça o cálculo do valor das seguintes métricas:

- Falta de coesão em métodos (LCOM4); e
- Resposta para uma classe (RFC).

O modificador `static` pode ser utilizado em C++ dentro do contexto de classes. Empregando esse modificador, tanto atributos quanto métodos podem ser modificados, levando a características especiais para esses membros da classe.

Informe a saída no terminal do programa em C++ abaixo que utiliza membros marcados com o modificador `static`.

```
01. #include <iostream>
02. class X{
03.     static int a;
04.     public:
05.     static int incA(){return a++;}
06.     int getA(){return a;}
07. };
08. int X::a = 10;
09. int main(){
10.     X x1;x1.incA();
11.     X x2;x2.incA();
12.     std::cout << x1.incA() << " ";
13.     X::incA();
14.     std::cout << x2.incA() << " ";
15.     std::cout << X::incA() << " ";
16.     std::cout << x1.getA() << " ";
17.     return 0;
18. }
```

Uma função membro virtual de uma classe em C++ é um método especial que é marcado com o modificador `virtual`. Considerando esse importante conceito da linguagem, informe a saída no terminal do seguinte programa em C++ que utiliza métodos virtuais.

```
01. #include <iostream>
02. struct X{
03.     int f(int a){return 4*a;}
04. };
05. struct Y:X{
06.     virtual int f(int a){return a^4;}
07. };
08. struct Z:Y{
09.     int f(int a){return a+4;}
10. };
11. int f(X *x){return x->f(4);}
12. int g(Y &y){return y.f(6);}
13. int main(){
14.     X x;Y y;Z z;
15.     std::cout << f(&x) << " " << f(&y) << " " << f(&z) << std::endl;
16.     std::cout << g(y) << " " << g(z) << std::endl;
17.     return 0;
18. }
```

Considere o programa abaixo. Identifique a *seção crítica* do programa indicando o(s) número(s) da(s) linha(s) correspondente(s). Justifique sua resposta.

```
01. #include <stdio.h>
02. #include <pthread.h>
03. #include <stdlib.h>
04. #define TAM_VET 100
05. #define N_THR 4
06. int vet[TAM_VET];
07. int total = 0;
08. void *contar_par(void *param);
09. int main(int argc, char **argv)
10. {
11.     int i;
12.     char i_str[N_THR][5];
13.     pthread_t tid[N_THR];
14.     for (i=0; i < TAM_VET; i++)
15.         vet[i] = rand();
16.     for (i=0; i < N_THR; i++) {
17.         sprintf(i_str[i], "%d", i);
18.         pthread_create(&tid[i], NULL, contar_par, i_str[i]);
19.     }
20.     for (i = 0; i < N_THR; i++)
21.         pthread_join(tid[i], NULL);
22.     printf("total = %d\n", total);
23.     exit(0);
24. }
25. void *contar_par(void *param) {
26.     int id = atoi(param);
27.     int i, inicio;
28.     inicio = id * (TAM_VET / N_THR);
29.     for (i=inicio; i < inicio + (TAM_VET / N_THR); i++)
30.         if ((vet[i] % 2) == 0)
31.             total++;
32.     pthread_exit(0);
33. }
```

**Observações:** Segue uma breve explicação na página seguinte sobre as funções da biblioteca pthread utilizadas no programa acima.

## 8ª QUESTÃO (CONTINUAÇÃO)

- `pthread_create(tid[i], NULL, contar_par, i_str[i]);`

Esta função cria um thread que irá executar a função `contar_par` e esta função terá como parâmetro `i_str[i]`. A identificação do thread criado será retornada em `tid[i]`.

- `pthread_join(tid[i], NULL);`

Esta função bloqueia o thread que a está chamando até que o thread com identificação `tid[i]` termine.

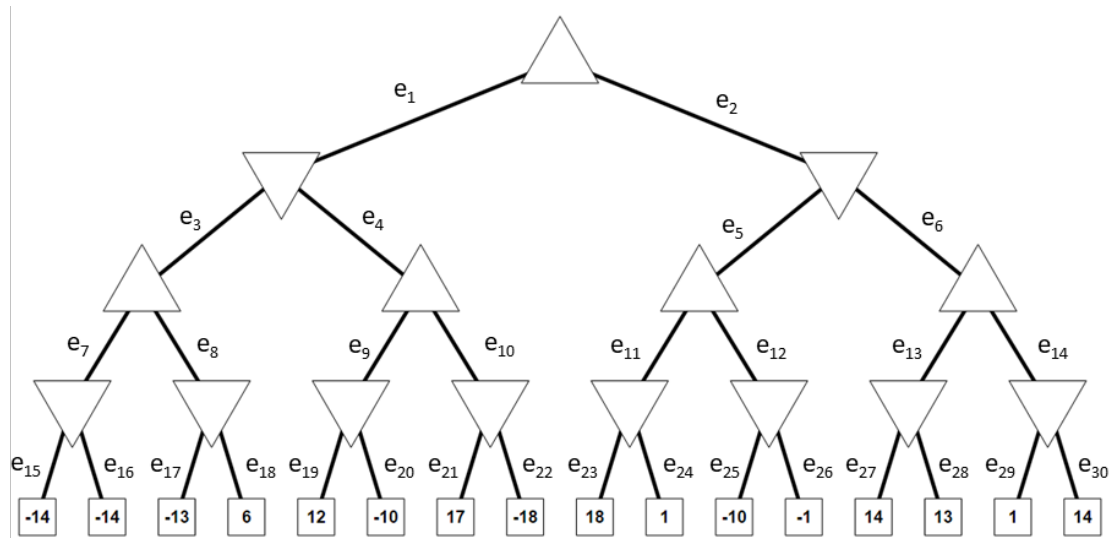
- `pthread_exit(0);`

Esta função termina a execução do thread que chamou a função retornando o valor 0 (zero).

9ª QUESTÃO

Valor: 1,0

Seja a árvore de busca, ilustrada na figura abaixo, de um jogo competitivo com dois jogadores racionais (MIN e MAX), onde o valor de cada folha representa o valor de utilidade da jogada correspondente. Considere MAX o jogador de referência.



Responda o que se pede, justificando:

- Indique o(s) número(s) da(s) aresta(s) que seria(m) podada(s) pelo algoritmo Minimax com Poda Alfa-Beta. Caso exista mais de uma aresta, apresente os números dessas arestas em ordem de ocorrência da poda (por exemplo,  $e_{65}, e_{82}, e_{70}$ ). Considere que a ordem de visita dos vértices deve ser sempre do vértice mais à esquerda para o vértice mais à direita.
- Indique a sequência de arestas escolhidas pelos jogadores nas circunstâncias indicadas acima.

10ª QUESTÃO

Valor: 1,0

Considere uma arquitetura de computadores genérica **MIPS** (Milhões de Instruções Por Segundo), que utiliza palavras de **32 bits** para dados e instruções. Além disso, essa arquitetura possui um conjunto de instruções reduzido, com instruções de **32 bits**. Nessa arquitetura, foram adicionados *buffers* entre o *hardware* das **5 (cinco)** etapas do ciclo de instrução, de tal modo que o conceito de *PIPELINE* está funcional.

Para esta questão, identifique todas as dependências de dados no código a seguir.

```
add $3, $4, $2
sub $5, $3, $1
lw $3, 200 ($6)
add $7, $3, $6
or $8, $5, $2
```

Utilize a sintaxe:

```
add $rd, $rs, $rt
lw $rd, offset ($rs)
```

onde *rd* é o registrador destino, *rs* e *rt* são os operandos da instrução.