

**TECNOLOGIA EM ANÁLISE E  
DESENVOLVIMENTO DE SISTEMAS**

**28**

**QUESTÃO DISCURSIVA 1**

**TEXTO I**

Em época de censura, a própria existência da arte passa a ser questionada. Surgem debates em jornais, na rua, em casa, para discutir sua relevância. Não podemos deixar de nos perguntar como chegamos a essa estranha situação em que precisamos justificar a própria existência da arte. Ela pode ser julgada apressadamente como boa ou ruim, mas nem por isso deixa de ser arte.

O cineasta franco-suíço Jean-Luc Godard aponta para o fato de que “a cultura é a regra; a arte é a exceção”. A arte é, dentro da cultura, o que tensiona a própria cultura para assim levá-la para outros lugares. Enquanto a cultura regula, a arte destoa e movimenta. A arte questiona, incomoda e transforma. Arte e cultura se contradizem, mas andam de mãos dadas.

Os psicanalistas Suely Rolnik e Félix Guattari consideram que o conceito de cultura é profundamente reacionário. É uma maneira de separar atividades semióticas em esferas, às quais os homens são remetidos. Tais atividades, assim isoladas, são padronizadas para o modo de semiotização dominante. A arte, por sua vez, existe plenamente quando junta o que é separado, questiona o que é geralmente aceito, grita onde há silêncio, desorganizando e reorganizando a cultura. Quando se discutem os limites da arte, são, na verdade, os limites da nossa tolerância que estão sendo debatidos.

SEROUSSI, B. O que faz a arte? In: OLIVIERE, C.; NATALE, E. (org.). **Direito, arte e liberdade**. São Paulo: Edições Sesc SP, 2018. p. 26-42 (adaptado).

**TEXTO II**

**Capítulo I  
Dos Direitos e Deveres Individuais e Coletivos**

Art. 5º Todos são iguais perante a lei, sem distinção de qualquer natureza, garantindo-se aos brasileiros e aos estrangeiros residentes no País a inviolabilidade do direito à vida, à liberdade, à igualdade, à segurança e à propriedade, nos termos seguintes:

[...]

IX - é livre a expressão da atividade intelectual, artística, científica e de comunicação, independentemente de censura ou licença.

BRASIL. Constituição Federal do Brasil. Disponível em: [https://www.senado.leg.br/atividade/const/con1988/con1988\\_15.12.2016/art\\_5\\_.asp](https://www.senado.leg.br/atividade/const/con1988/con1988_15.12.2016/art_5_.asp). Acesso em: 2 maio 2020.

Considerando as informações e os argumentos presentes nos textos I e II, discorra a respeito da relação entre arte, cultura e censura, à luz da ideia de liberdade artística garantida pela Constituição Federal de 1988. Apresente, em seu texto, duas ações educativas que podem contribuir para minimizar essas tensões e garantir a liberdade artística prevista pela lei. (valor: 10,0 pontos)

## PADRÃO DE RESPOSTA

---

O respondente deve, a partir dos argumentos presentes no texto I, refletir sobre as tensões existentes entre a arte e a cultura no Brasil contemporâneo e sobre a liberdade artística explicitado no artigo 5º da Constituição Federal (Texto II), de modo a perceber a ilegitimidade dos movimentos de censura que tem eclodido em determinados segmentos da sociedade brasileira.

O respondente deve, ainda, apresentar duas ações educativas para a superação das tensões citadas, como: encontros de artistas e público em escolas e outros espaços públicos; projetos de visitação a espaços culturais, como museus e galerias, voltados para a formação de público/plateia; debates em espaços públicos a respeito da liberdade artística, etc.

(Valor: 10,0 pontos)

## QUESTÃO DISCURSIVA 2

---

---

### TEXTO I

Uma cidade é considerada inteligente quando: i) nela se utiliza a tecnologia para melhorar a sua infraestrutura e seus serviços, tornando os setores de administração, educação, saúde, segurança pública, moradia e transporte mais inteligentes, interconectados e eficientes, beneficiando toda a população; e ii) está comprometida com o meio ambiente e com sua herança histórica e cultural.

AQUINO, A. L. L. et al. Cidades inteligentes, um novo paradigma da sociedade do conhecimento. **Blucher Education Proceedings**, v. 1, n. 1, p. 165-178, 2015 (adaptado).

### TEXTO II

A evolução para uma cidade mais inteligente, mais integrada, mais inovadora pressupõe uma visão holística e sistêmica do espaço urbano e a integração efetiva dos vários atores e setores. Para tal, é necessário ir além dos investimentos em inovação tecnológica e inovar também na gestão, no planejamento, no modelo de governança e no desenvolvimento de políticas públicas.

CAMPOS, C. C. et al. Cidades inteligentes e mobilidade urbana. **Cadernos FGV Projetos**, n. 24, 2014 (adaptado).

A partir do conceito de cidade inteligente exposto nos textos, faça o que se pede nos itens a seguir.

- Explique de que modo as cidades inteligentes podem contribuir para a melhoria das questões relacionadas ao desenvolvimento sustentável. (valor: 5,0 pontos)
- Apresente uma proposta de intervenção urbana que pode gerar impacto social e contribuir para a melhoria da vida em comunidade. (valor: 5,0 pontos)

## PADRÃO DE RESPOSTA

---

- O respondente deve mencionar que as cidades inteligentes podem diminuir o impacto ambiental dos aglomerados urbanos, pois, ao utilizarem a tecnologia como um fator indispensável para modernizar e oferecer melhor infraestrutura e serviços, colaboram, por exemplo, com a redução no consumo de energia e na emissão de CO<sub>2</sub>.
- O respondente deve elaborar uma proposta de intervenção que gere impacto social e contribua para a melhoria da vida em comunidade. Exemplos de intervenção incluem:
  - ✓ Proposição de aplicativos para:
    - compartilhamento de transporte (caronas);
    - oferecimento de pequenos serviços (babá, pet sitter, acompanhamento de idosos, acompanhamento psicológico);
    - doação de produtos, alimentos, etc.

- ✓ Plano de ação a fim de oferecer serviços específicos a grupos menos favorecidos, como idosos ou população de rua.
- ✓ Concepção de artefatos urbanos para melhorar a mobilidade urbana ou para permitir a passagem de fauna.

Etc.

### QUESTÃO DISCURSIVA 3

Considere a realização de uma pesquisa com 1 000 pessoas para obtenção das seguintes informações: o valor da maior altura; o valor da menor altura; a média das alturas; quantas pessoas têm altura inferior à média das alturas.

Considere, ainda, que um programador foi selecionado para desenvolver um modelo de código que soluciona o problema automatizando a coleta das alturas e a geração das informações.

Com base nas informações apresentadas, desenvolva o código adequado para resolver o problema usando pseudocódigo ou uma linguagem de programação. (valor: 10,0 pontos)

#### PADRÃO DE RESPOSTA

O respondente deve desenvolver o seguinte código:

algoritmo "pesquisa"

var altura[1000], menor, maior, total=0, media:real

var qtdmenor=0, ctpessoas: inteiro

inicio

leia altura[1]

menor = altura[1]

maior = altura[1]

total = altura[1]

para ctpessoas=2 até 1000 faça

leia (altura[ctpessoas])

total=total + altura[ctpessoas]

fimpara

media = total/1000

para ctpessoas=1 até 1000 faça

se menor > altura[ctpessoas] então menor = altura[ctpessoas]

se maior < altura[ctpessoas] então maior = altura[ctpessoas]

se media > altura[ctpessoas] então qtdmenor = qtdmenor + 1

fimpara

escreva ("Maior = ", maior, " Menor= ",menor," Média= ",media, " Quantidade de alturas menores que a média= ",qtdmenor)

## QUESTÃO DISCURSIVA 4

Um analista de requisitos identificou as seguintes histórias de usuários para um sistema web de reserva de passagens de uma empresa aérea:

HST 01: COMO cliente, GOSTARIA de me cadastrar no site da empresa informando meu e-mail, nome, CPF, telefone e endereço PARA poder usar os serviços web da empresa.

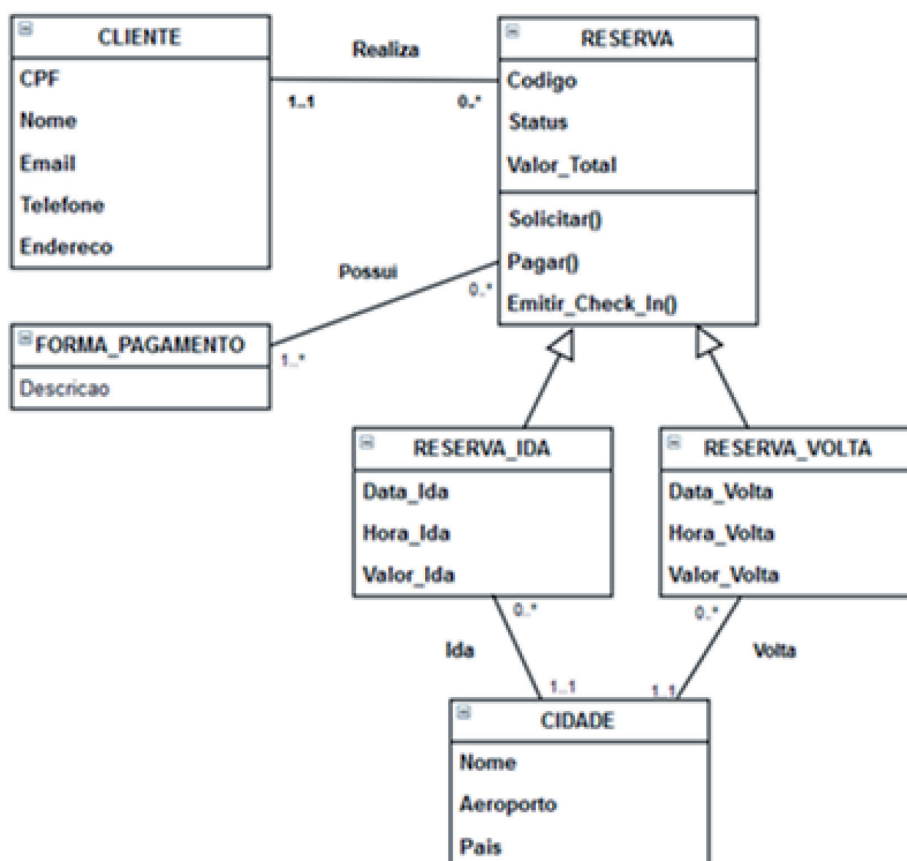
HST 02: COMO cliente, GOSTARIA de pesquisar por preços das passagens aéreas informando data, hora, cidade e aeroporto de ida e de volta PARA poder escolher uma opção de reserva que me interessa.

HST 03: COMO cliente, GOSTARIA de confirmar uma reserva de passagem aérea selecionada, escolhendo uma das formas de pagamento disponibilizada pela empresa PARA poder viajar para o local escolhido na reserva.

HST 04: COMO cliente, GOSTARIA de fazer o *check-in* online da reserva PARA poder realizar meu embarque. Tendo por base essas histórias de usuários, elabore um diagrama de classes da UML (*Unified Modeling Language*) para o sistema, identificando as classes (máximo seis), atributos e associações entre as classes com as respectivas multiplicidades. Deverão ser representadas as reservas de ida e de volta como classes de uma associação do tipo generalização (herança) e também ser identificados pelo menos três métodos para a classe Reserva. Para as demais classes não é necessário identificar os métodos. Também não há necessidade de identificar a multiplicidade dos atributos nas classes. (valor: 10,0 pontos).

### PADRÃO DE RESPOSTA

O respondente deve elaborar um diagrama da UML para o sistema, contendo no mínimo, as classes, atributos, métodos e associações a seguir:



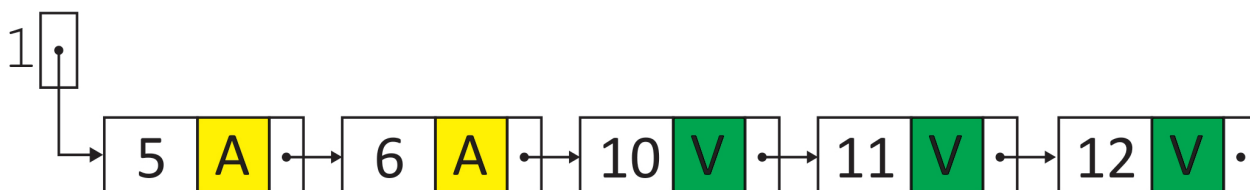
## QUESTÃO DISCURSIVA 5

Com a finalidade de melhorar o atendimento e priorizar os casos mais urgentes, a direção de um hospital criou um sistema de triagem em que um profissional da saúde classifica a ordem de atendimento com base numa avaliação prévia do paciente, entregando-lhe um cartão numerado verde (V) ou amarelo (A), que define o menor ou maior grau de urgência da ocorrência, respectivamente. Para informatizar esse processo, o software desenvolvido tem como base o seguinte trecho de código-fonte:

```
1 struct lista {
2     int numero;
3     char cor;
4     struct lista* prox;
5 };
6
7 typedef struct lista Lista;
8
9 Lista* inserir_fim(Lista* l, Lista* no) {
10     Lista* aux;
11
12     no->prox = NULL;
13     aux = l;
14     while (aux->prox != NULL)
15         aux = aux->prox;
16     aux->prox = no;
17
18     return l;
19 }
20
21 Lista* inserir(Lista* l, int numero, char cor) {
22     Lista* no = (Lista*) malloc(sizeof(Lista));
23     no->numero = numero;
24     no->cor = cor;
25
26     if (l == NULL) {
27         no->prox = l;
28         l = no;
29     } else {
30         if (no->cor == 'V')
31             l = inserir_fim(l, no);
32         else
33             l = inserir_prioridade(l, no);
34     }
35
36     return l;
37 }
```

Na linha 21, a função `inserir` recebe o número e a cor do cartão entregue ao paciente na triagem. Pacientes com cartão verde são inseridos no final da fila pela função `inserir_fim` (linhas 9-19). Pacientes com cartão amarelo têm prioridade no atendimento e são inseridos no início da fila, em

ordem de chegada, pela função `inserir_prioridade`. Portanto, se são entregues os cartões 10-V, 11-V, 5-A, 12-V e 6-A, nessa ordem, a fila deve ficar assim organizada:



Considerando o processo de triagem descrito e o trecho de código-fonte apresentado, implemente a função `inserir_prioridade` conforme indicado.

```
Lista* inserir_prioridade(Lista* l, Lista* no)
```

(valor: 10,0 pontos)

## **PADRÃO DE RESPOSTA**

O respondente deve implementar a função `inserir_prioridade` no código fonte apresentado. Assim, a lógica do algoritmo deve atender a dois fundamentos básicos de inserção de nós em uma lista simplesmente encadeada.

- 1) **INSERÇÃO NO INÍCIO:** caso ainda não haja nenhum cartão amarelo na fila, o novo nó é inserido no início, como primeiro elemento da fila;
- 2) **INSERÇÃO NO MEIO:** se já houver cartões amarelos na fila, o novo nó é inserido no meio ou no final da fila, a depender, respectivamente, da presença ou da ausência de algum cartão verde; para ambos os casos, contudo, a lógica de inserção do nó é exatamente a mesma, pois as variáveis `ant` e `aux` são usadas para percorrer a lista simultaneamente e atualizar as referências (ponteiros) entre os nós.

É natural que o estudante siga essa sequência lógica e, portanto, espera-se que escreva a função

```
1 Lista* inserir_prioridade(Lista* l, Lista* no) {
2     if (l->cartao == 'V') {
3         no->prox = l;
4         l = no;
5     } else {
6         Lista* ant = l;
7         Lista* aux = ant->prox;
8         while (aux != NULL && aux->cartao == 'A') {
9             ant = aux;
10            aux = aux->prox;
11        }
12        ant->prox = no;
13        no->prox = aux;
14    }
15
16    return l;
17 }
```

**Observação:** Serão aceitas respostas em outra linguagem de programação compatível ou pseudo-código.