



CIÊNCIA DA COMPUTAÇÃO

MATERIAL INSTRUCIONAL ESPECÍFICO

TOMO 2

CQA - COMISSÃO DE QUALIFICAÇÃO E AVALIAÇÃO

CIÊNCIA DA COMPUTAÇÃO

MATERIAL INSTRUCIONAL ESPECÍFICO

TOMO 2

Material instrucional específico, cujo conteúdo integral ou parcial não pode ser reproduzido ou utilizado sem autorização expressa, por escrito, da CQA/UNIP – Comissão de Qualificação e Avaliação da UNIP – UNIVERSIDADE PAULISTA.

Questão 1

Questão 1.¹

Considere a relação EMPREGADO (NumeroEmp, RG, nome, sobrenome, salario, endereço), em que o atributo grifado corresponde à chave primária da relação. Suponha que se deseje realizar as seguintes consultas:

- 1 Listar o nome dos empregados com sobrenome Silva;
- 2 Listar o nome dos empregados em ordem crescente de seus sobrenomes.

Em relação à definição de um índice sobre o atributo sobrenome para melhorar o desempenho das consultas acima, julgue os itens a seguir.

- I. Um índice que implemente Árvore-B+ será adequado para melhorar o desempenho da consulta 1.
- II. Um índice que implemente Árvore-B+ será adequado para melhorar o desempenho da consulta 2.
- III. Um índice que implemente uma função hash será adequado para melhorar o desempenho da consulta 1.
- IV. Um índice que implemente uma função hash será adequado para melhorar o desempenho da consulta 2.

Assinale a opção correta.

- A. Apenas um item está certo.
- B. Apenas os itens I e II estão certos.
- C. Apenas os itens III e IV estão certos.
- D. Apenas os itens I, II e III estão certos.
- E. Todos os itens estão certos.

1. Introdução teórica

1.1. Índices

Índices são organizações de dados que permitem acelerar o tempo de acesso às linhas de uma tabela.

¹Questão 21 - Enade 2008.

Há dois tipos principais de índices: os ordenados, baseados na ordenação de valores, e os *hash*, baseados na distribuição uniforme de valores por meio de uma faixa de *buckets* (páginas de dados). O *bucket* ao qual dado valor é atribuído é determinado pela função *hash*.

1.1.1. Índices ordenados

Os registros em um arquivo indexado podem ser armazenados seguindo alguma ordem. O quadro 1 apresenta um índice primário formado pela chave primária da tabela, representada pelo campo FUNCIONAL, e por dois índices secundários, formados pelos campos NOME e SOBRENOME. Os dados estão organizados obedecendo à ordem FUNCIONAL → NOME → SOBRENOME, sendo que os campos restantes também poderiam ser utilizados como índices.

Quadro 1. Registros organizados de forma indexada.

FUNCIONAL	NOME	SOBRENOME	CARGO	SALÁRIO
101	João	Junqueira	Gerente	20.000,00
102	João	Silva	Analista Junior	2.500,00
103	João	Teixeira	Analista Sênior	10.000,00
104	Manoel	Aparecido	Analista Pleno	8.000,00
105	Manoel	Silva	Analista Negócio	7.000,00

No caso do quadro 2, a chave utilizada para índice foi o campo CARGO.

Quadro 2. Registros organizados de forma indexada – chave (CARGO).

FUNCIONAL	NOME	SOBRENOME	CARGO	SALÁRIO
102	João	Silva	Analista Junior	2.500,00
105	Manoel	Silva	Analista Negócio	7.000,00
104	Manoel	Aparecido	Analista Pleno	8.000,00
103	João	Teixeira	Analista Sênior	10.000,00
101	João	Junqueira	Gerente	20.000,00

Vejamos, a seguir, uma breve descrição dos índices.

- **Índices primários:** os arquivos estão ordenados sequencialmente por alguma chave de procura (em geral, a própria chave primária).
- **Índices secundários:** as chaves de pesquisa especificam uma ordem diferente da ordem sequencial do arquivo indexado.
- **Índices densos:** um registro do índice aparece para cada registro do arquivo. No exemplo da figura 1, na procura dos registros da agência Paraíso utilizando um índice denso, segue-se o ponteiro diretamente para o primeiro registro com agência Paraíso. Processa-se esse registro e segue-se o ponteiro do registro para localizar o próximo na ordem da chave de procura (Agencia_Bancaria). Continua-se o processamento até que se encontre um registro que não pertença à agência Paraíso.

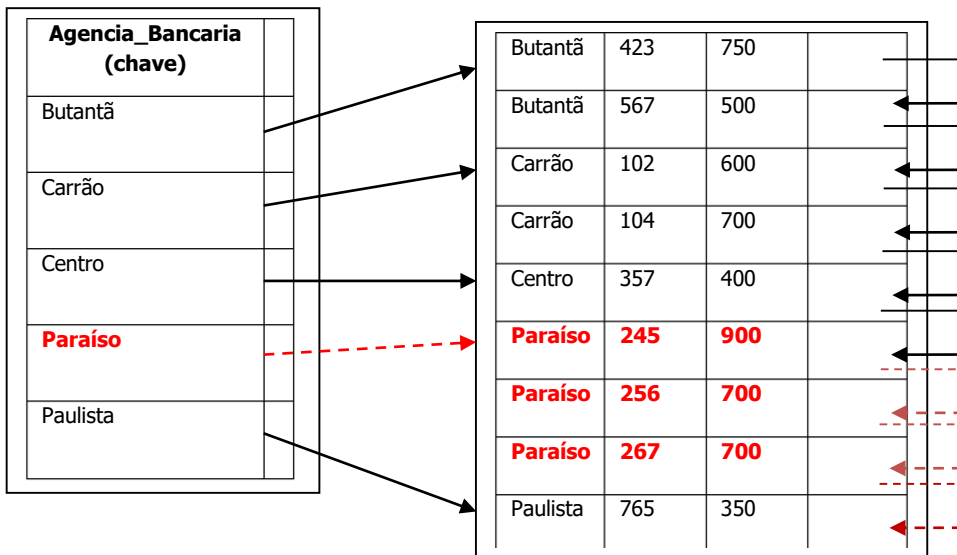


Figura 1. Índice denso.

- **Índices esparsos:** um registro de índice é criado apenas para alguns dos valores. Assim como nos índices densos, cada registro do índice esparsos contém um valor de chave de procura e um ponteiro para o primeiro registro de dados com esse valor de chave.

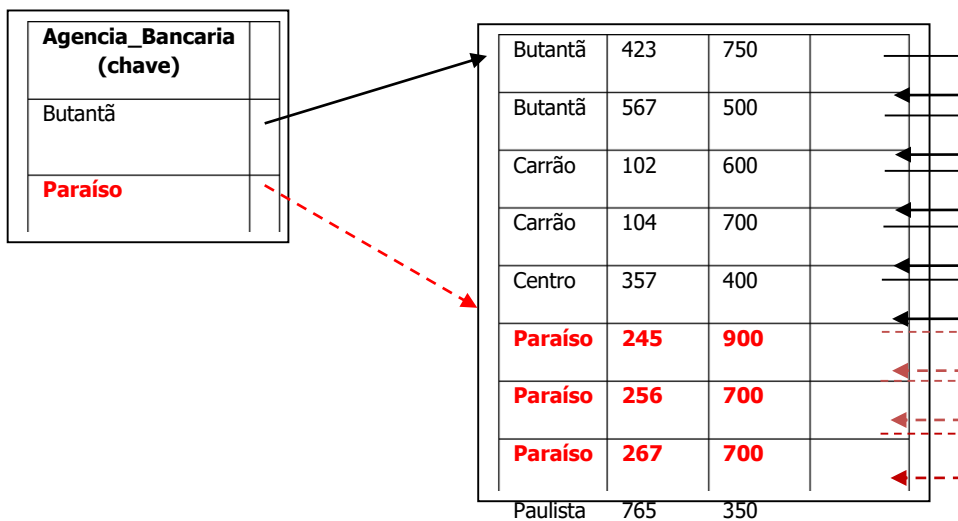


Figura 2. Índice esparsos.

- **Índices completos e parciais:** os índices completos indexam todas as linhas de determinada tabela, enquanto os parciais indexam somente um subconjunto de linhas definidas por um atributo durante o processo de criação do índice.
- **Índices de níveis múltiplos:** são definidos por mais de uma coluna. Devem ser utilizados em conjunto com o operador lógico AND.

1.2. Estruturas de dados utilizadas na implementação de índices

A principal desvantagem da organização de índices sequenciais está relacionada à degradação do seu desempenho conforme o arquivo sofre processos de inserção ou de remoção de registros. Isso ocorre em decorrência das procuras nos índices e das varreduras sequenciais de dados. Essa degradação pode ser remediada pela reorganização do arquivo, mas reorganizações frequentes não são desejáveis.

Nos bancos de dados relacionais, as estruturas de dados mais usuais são as árvores, indicadas em consultas por igualdade, desigualdade e ordenação. Nos índices criados por meio de *hash*, os valores-chave do índice não se encontram organizados e seu desempenho é melhor quando os valores chave de índice são conhecidos. Esse procedimento utiliza o valor-chave no cálculo do endereço do dado desejado pela utilização da função *hash*, que pode otimizar o acesso a um dado somente em consultas por igualdade.

2. Indicações bibliográficas

- SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. *Sistemas de banco de dados*. São Paulo: Makron Books, 2003.
- ZIVIANI, N. *Projeto de algoritmos com implementação em Pascal e C*. 5. ed. São Paulo: Pioneira, 1999.

3. Análise das afirmativas

I – Afirmativa correta.

JUSTIFICATIVA. A consulta utiliza um filtro de igualdade e as Árvores-B são indicadas para esse tipo de consulta, conforme mostramos a seguir.

```
SELECT    Nome
FROM      Empregado
WHERE     Sobrenome = 'Silva';
```

II – Afirmativa correta.

JUSTIFICATIVA. A consulta utiliza um filtro de ordenação e as Árvores-B são indicadas para esse tipo de consulta, conforme mostramos a seguir.

```
SELECT      Nome
FROM        Empregado
ORDER BY    Sobrenome      ASC;
```

III – Afirmativa correta.

JUSTIFICATIVA. A consulta utiliza um filtro de igualdade e as funções *hash* são indicadas para esse tipo de consulta.

IV – Afirmativa incorreta.

JUSTIFICATIVA. A consulta utiliza um filtro de ordenação e as funções *hash* não são indicadas para esse tipo de consulta.

Alternativa correta: D.

Questão 2

Questão 2.²

Qual tipo de software tradutor deve ser utilizado para programas em geral, quando a velocidade de execução é uma exigência de alta prioridade?

- A. Compiladores
- B. Interpretadores
- C. Tradutores híbridos
- D. Macroprocessadores
- E. Interpretadores de macroinstruções

1. Introdução teórica

1.1. Programas de computadores

Os programas de computadores são construídos em linguagens responsáveis pela especificação dos processos computacionais. Os computadores não executam os textos desses programas diretamente, mas sim em sequências específicas de instruções. O texto deve ser traduzido em grupos de instruções, em ordem determinada, para que possa ser reconhecido e executado pela máquina.

1.2. Processador de linguagem

Processadores de linguagem são programas complexos que traduzem os comandos escritos pelos usuários em uma linguagem que possa ser reconhecida pelo computador.

Os processadores são classificados em:

- interpretadores (interpretam e executam diretamente os comandos escritos pelo usuário);
- tradutores (traduzem o texto fonte em uma linguagem alvo).

Podemos dividir os tradutores em montadores e compiladores, os quais traduzem, respectivamente, linguagens de baixo nível e de alto nível (figura 1).

²Questão 22 - Enade 2008.

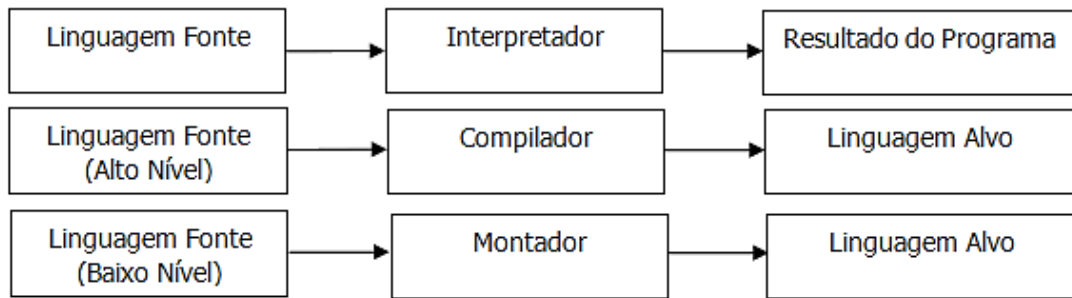


Figura 1. Processadores de linguagens.

2. Indicação bibliográfica

- ALFRED, V. A. *et al. Compiladores: princípios, técnicas e ferramentas*. 2. ed. São Paulo: Pearson Addison Wesley, 2008.

3. Análise das alternativas

A – Alternativa correta.

JUSTIFICATIVA. Um compilador é um programa que recebe como entrada um programa em linguagem de programação (linguagem fonte) e o traduz para um programa equivalente em outra linguagem (linguagem objeto), conforme indicado na figura 2. Se o programa objeto estiver em linguagem de máquina executável, poderá ser chamado pelo usuário para processar entradas e produzir saídas. O programa executável resultante da compilação não apresenta nenhum tipo de dependência com o compilador. A execução do programa compilado consiste em seu carregamento na memória e posterior efetivação do seu fluxo de execução.

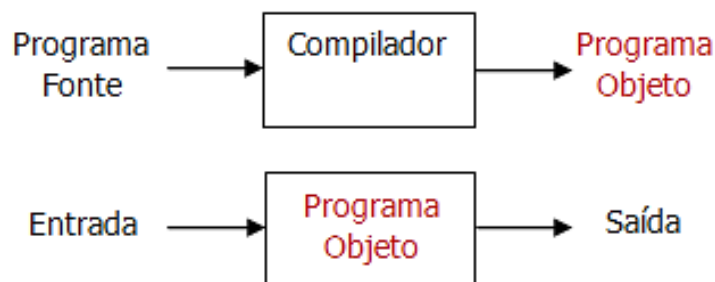


Figura 2. Processadores de linguagens.

O programa objeto em linguagem de máquina produzido por um compilador normalmente é mais rápido do que um interpretador.

B – Alternativa incorreta.

JUSTIFICATIVA. Um interpretador, ao contrário dos compiladores, não produz um programa objeto como resultado da tradução. Ele executa diretamente as operações especificadas no programa fonte sobre as entradas fornecidas pelo usuário. Como o interpretador não gera um programa executável, o programa fonte somente poderá ser executado na presença do interpretador, gerando um vínculo de dependência (figura 3). Os programas interpretados são mais lentos do que os programas compilados.

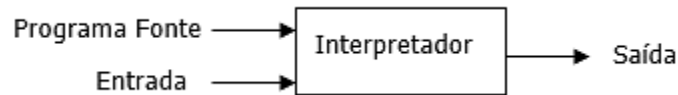


Figura 3. Interpretador.

Os interpretadores oferecem melhor diagnóstico de erro do que os compiladores, pois executam o programa fonte instrução por instrução.

C – Alternativa incorreta.

JUSTIFICATIVA. Alguns processadores de linguagem, como o utilizado pela linguagem Java, combinam compilação e interpretação (figura 4). Um programa fonte em Java pode ser compilado para uma forma intermediária, chamada bytecode, e, em seguida, sofrer um processo de interpretação dos bytecodes por uma máquina virtual (software capaz de emular o hardware e o sistema operacional do computador). Essa característica permite que a linguagem Java seja multiplataforma. Nesse caso, o programa passará por duas etapas, compilação e interpretação. Portanto, os compiladores são mais eficientes.

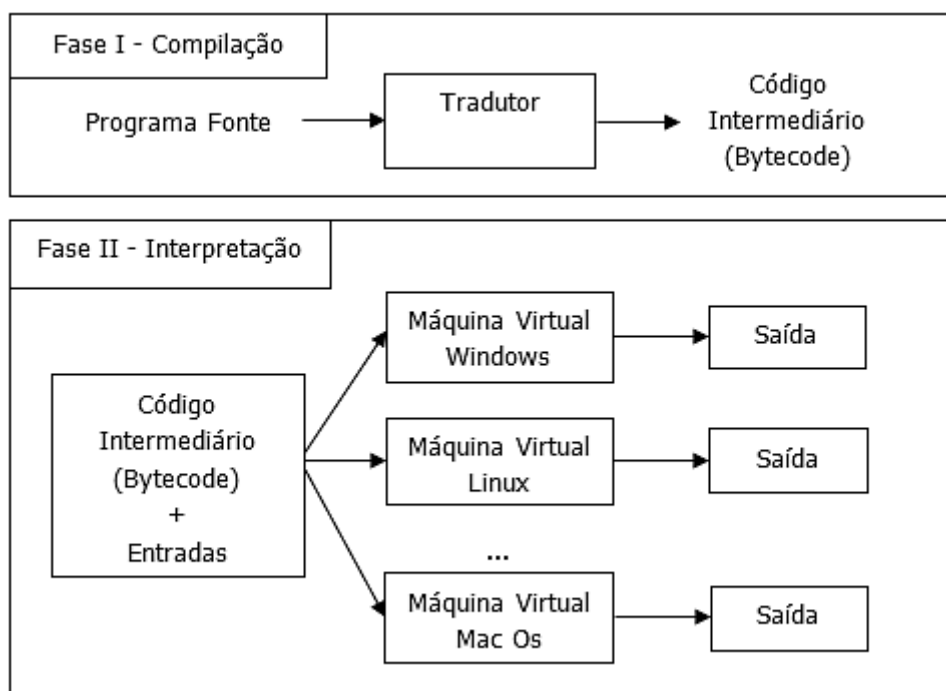


Figura 4. Compilação e interpretação.

D – Alternativa incorreta.

JUSTIFICATIVA. Os macroprocessadores traduzem programas escritos em linguagem de alto nível em outros programas que também utilizam linguagem de alto nível. Esse modelo não atende aos requisitos da questão.

E – Alternativa incorreta.

JUSTIFICATIVA. Os interpretadores de macroinstruções não atendem aos requisitos da questão, pois são conjuntos de instruções que podem ser utilizados como uma única instrução ao longo de um programa em linguagem de montagem.

Questão 3

Questão 3.³

Considere o esquema de banco de dados relacional apresentado a seguir, formado por 4 relações, que representa o conjunto de estudantes de uma universidade que podem, ou não, morar em repúblicas (moradias compartilhadas por estudantes). A relação *Estudante* foi modelada como um subconjunto da relação *Pessoa*.

Considere que os atributos grifados correspondam à chave primária da respectiva relação e os atributos que são seguidos da palavra *referencia* sejam chaves estrangeiras.

```
Pessoa(IdPessoa:integer,      Nome:varchar(40),
Endereco:varchar(40))
FonePessoa(IdPessoa:integer referencia Pessoa,
DDD:varchar(3), Prefixo:char(4), Nro:char(4))
Republica(IdRep:integer,      Nome:varchar(30),
Endereco:varchar(40))
Estudante(RA:integer,      Email:varchar(30),
IdPessoa:integer referencia Pessoa,
IdRep:integer referencia Republica)
```

Suponha que existam as seguintes tuplas no banco de dados:

Pessoa(1, 'José Silva', 'Rua 1, 20');

Republica(20, 'Várzea', 'Rua Chaves, 2001')

Qual opção apresenta apenas tuplas válidas para esse esquema de banco de dados relacional?

- A. Estudante(10, 'jsilva@ig.com.br', null, 20); FonePessoa(10, '019', '3761', '1370')
- B. Estudante(10, 'jsilva@ig.com.br', 1, null); FonePessoa(10, '019', '3761', '1370')
- C. Estudante(10, 'jsilva@ig.com.br', 1, 20); FonePessoa(1, null, '3761', '1370')
- D. Estudante(10, 'jsilva@ig.com.br', 1, 50); FonePessoa(1, '019', '3761', '1370')
- E. Estudante(10, 'jsilva@ig.com.br', 1, null); FonePessoa(1, '019', '3761', '1370')

³Questão 23 - Enade 2008.

1. Introdução teórica

1.1. Teoria relacional

A abordagem relacional descreve um banco de dados por meio da teoria dos conjuntos. Para os usuários, um banco de dados pode ser representado por um conjunto de tabelas bidimensionais, originadas em linhas e colunas.

Os dados são as estruturas mais primitivas de um banco de dados. Sua integridade influencia de forma decisiva na qualidade da informação produzida.

Em um banco de dados, as tabelas ou relações são responsáveis pelo armazenamento dos dados. Ao se planejar a estrutura de uma relação, deve-se pensar no que se pretende guardar, no espaço necessário e nas regras de integridade.

1.2. Regras de integridade (constraints)

As regras de integridade estão descritas a seguir.

- **Chave Primária (Primary Key):** corresponde à coluna ou ao conjunto de colunas que identifica unicamente cada linha na tabela. Cada tabela tem uma Primary Key, que deve ser única. As colunas de uma Primary Key composta devem ser únicas em combinação e, individualmente, podem ter duplicações, mas nunca em combinação. Nenhuma parte de uma Primary Key pode ser null.
- **Chave Estrangeira (Foreign Key):** corresponde à coluna ou à combinação de colunas de uma tabela que referencia uma coluna em outra tabela ou na mesma tabela (autorrelacionamento). O valor de uma Foreign Key deve existir em outra tabela ou ser nulo.
- **Não Nulo (NOT NULL):** ao se determinar que um campo é not null, obriga-se a digitação de um dado por parte do usuário.
- **Único (UNIQUE):** garante que um dado não terá o seu valor replicado entre os registros.
- **Faixa de valores (CHECK):** permite estabelecer uma faixa de valores válidos.

2. Indicação bibliográfica

- SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. *Sistemas de banco de dados*. São Paulo: Makron Books, 2003.

3. Análise das alternativas

Para analisarmos as alternativas da questão, vamos considerar, inicialmente, os quadros abaixo.

Pessoa		
<u>IdPessoa</u>	Nome	Endereco
1	Jose Silva	Rua 1, 20

FonePessoa			
<u>IdPessoa</u>	DDD	Prefixo	Nro

Republica		
<u>IdRep</u>	Nome	Endereco
20	Várzea	Rua Chaves, 2001

Estudante			
<u>Ra</u>	Email	<u>IdPessoa</u>	<u>IdRep</u>

A – Alternativa incorreta.

JUSTIFICATIVA. A questão apresenta um erro de chave estrangeira entre IdPessoa da tabela FonePessoa e a tabela Pessoa, não sendo possível atribuir um telefone a uma pessoa que não existe. No caso da tabela Estudante, é possível haver uma chave estrangeira (IdPessoa) com valor null (veja os quadros a seguir).

Pessoa		
<u>IdPessoa</u>	Nome	Endereco
<u>1</u>	Jose Silva	Rua 1, 20

FonePessoa			
<u>IdPessoa</u>	<u>DDD</u>	<u>Prefixo</u>	<u>Nro</u>
<u>10</u>	019	3761	1370

Republica		
<u>IdRep</u>	Nome	Endereco
20	Várzea	Rua Chaves, 2001

Estudante			
<u>Ra</u>	Email	<u>IdPessoa</u>	<u>IdRep</u>
10	jsilva@ig.com.br	null	20

B – Alternativa incorreta.

JUSTIFICATIVA. A questão apresenta um erro de chave estrangeira entre IdPessoa da tabela FonePessoa e a tabela Pessoa, não sendo possível atribuir um telefone a uma pessoa que não existe. No caso da tabela Estudante, é possível haver uma chave estrangeira (IdRep) com valor null (veja os quadros a seguir).

Pessoa			FonePessoa			
<u>IdPessoa</u>	Nome	Endereco	<u>IdPessoa</u>	<u>DDD</u>	<u>Prefixo</u>	<u>Nro</u>
<u>1</u>	Jose Silva	Rua 1, 20	<u>10</u>	019	3761	1370

Republica			Estudante			
<u>IdRep</u>	Nome	Endereco	<u>Ra</u>	Email	<u>IdPessoa</u>	<u>IdRep</u>
20	Várzea	Rua Chaves, 2001	10	jsilva@ig.com.br	1	null

C – Alternativa incorreta.

JUSTIFICATIVA. Na tentativa de atribuirmos null ao campo DDD ocorre um erro de integridade na chave primária, que não pode ter nenhum valor nulo. As chaves estrangeiras IdPessoa e IdRep estão corretas, pois os valores atribuídos existem nos campos das tabelas referenciadas IdPessoa (Pessoa) e IdRep (Republica). Verifique os quadros abaixo.

Pessoa			FonePessoa			
<u>IdPessoa</u>	Nome	Endereco	<u>IdPessoa</u>	<u>DDD</u>	<u>Prefixo</u>	<u>Nro</u>
1	Jose Silva	Rua 1, 20	1	<u>null</u>	3761	1370

Republica			Estudante			
<u>IdRep</u>	Nome	Endereco	<u>Ra</u>	Email	<u>IdPessoa</u>	<u>IdRep</u>
20	Várzea	Rua Chaves, 2001	10	jsilva@ig.com.br	1	20

D – Alternativa incorreta.

JUSTIFICATIVA. Ao atribuímos o valor 50 ao campo IdRep (chave estrangeira) da tabela Estudante, houve quebra de integridade, pois o valor 50 não existe na tabela Republica (IdRep). Verifique os quadros abaixo.

Pessoa			FonePessoa			
<u>IdPessoa</u>	Nome	Endereco	<u>IdPessoa</u>	<u>DDD</u>	<u>Prefixo</u>	<u>Nro</u>
1	Jose Silva	Rua 1, 20	1	019	3761	1370

Republica			Estudante			
<u>IdRep</u>	Nome	Endereco	<u>Ra</u>	Email	<u>IdPessoa</u>	<u>IdRep</u>
20	Várzea	Rua Chaves, 2001	10	jsilva@ig.com.br	1	<u>50</u>

E – Alternativa correta.

JUSTIFICATIVA. Essa alternativa é a única que não apresenta quebra de integridade dos dados (vide quadros a seguir).

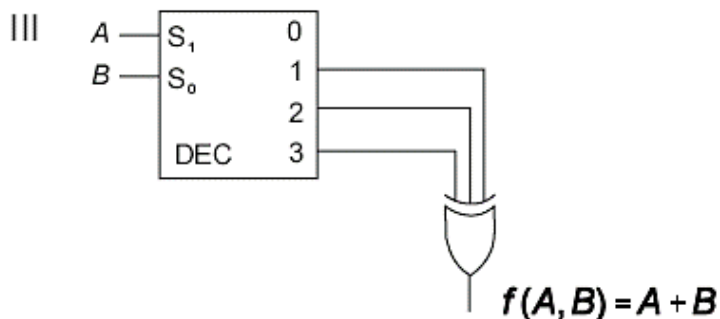
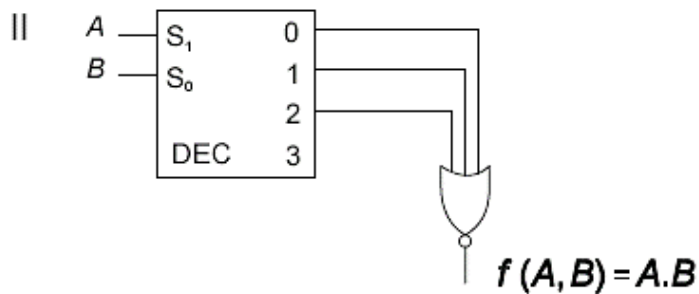
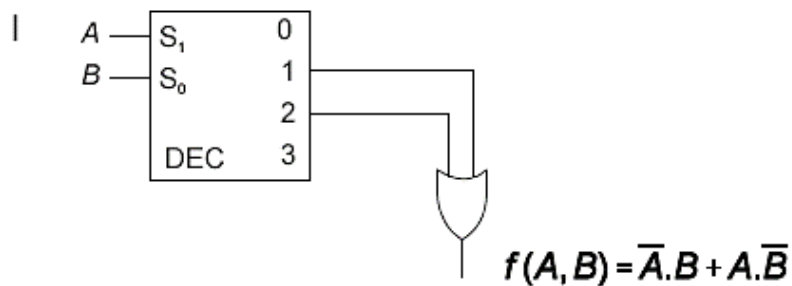
Pessoa			FonePessoa			
<u>IdPessoa</u>	Nome	Endereco	<u>IdPessoa</u>	DDD	Prefixo	Nro
1	Jose Silva	Rua 1, 20	1	019	3761	1370

Republica			Estudante			
<u>IdRep</u>	Nome	Endereco	<u>Ra</u>	Email	<u>IdPessoa</u>	<u>IdRep</u>
20	Várzea	Rua Chaves, 2001	10	jsilva@ig.com.br	1	null

Questão 4**Questão 4.⁴**

entradas		saídas			
S_1	S_0	0	1	2	3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Considere o bloco decodificador ilustrado acima, o qual opera segundo a tabela apresentada. Em cada item a seguir, julgue se a função lógica mostrada corresponde ao circuito lógico a ela associado.



⁴Questão 24 - Enade 2008.

Assinale a opção correta.

- A. Apenas um item está certo.
- B. Apenas os itens I e II estão certos.
- C. Apenas os itens I e III estão certos.
- D. Apenas os itens II e III estão certos.
- E. Todos os itens estão certos.

1. Introdução teórica

1.1. Operações lógicas básicas

O termo lógica é aplicado a circuitos digitais usados para implementar funções lógicas. Suas três operações básicas, NOT, AND e OR, estão descritas abaixo.

- NOT: comuta de um nível lógico para o nível lógico oposto.
- AND: gera uma saída verdadeira somente se as entradas forem verdadeiras.
- OR: gera uma saída falsa somente se as entradas forem falsas.

Um circuito que executa uma operação lógica é denominado porta lógica (figura 1).

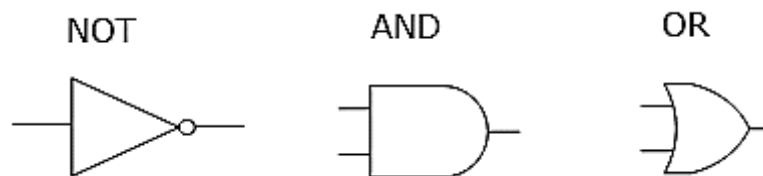


Figura 1. Portas lógicas.

As portas AND e OR podem ter um número maior do que 1 qualquer de entradas com uma saída. Em operações lógicas, as condições verdadeiro e falso são representadas por alto (1 - verdadeiro) e baixo (0 - falso).

1.2. Funções lógicas básicas

Os três elementos lógicos básicos AND, OR e NOT podem ser combinados para formar circuitos lógicos mais complexos que realizam as operações de comparação, aritmética, conversão de código, codificação, decodificação, seleção de dados, armazenamento e contagem. Essas operações estão descritas a seguir.

- Comparação: compara dois números e indica se são iguais ou não.

- Aritmética: a adição é realizada pelo somador; a subtração é realizada pelo subtrator; a multiplicação é realizada pelo multiplicador; e a divisão pode ser realizada por meio de uma série de subtrações, comparações e deslocamentos (associação de circuitos).
- Conversão de código: converte uma informação codificada de uma forma em outra forma de código como, por exemplo, a conversão de um número binário em decimal.
- Codificação: converte uma informação em alguma forma codificada como, por exemplo, a conversão de um dos dígitos decimais em um código binário (figura 2).

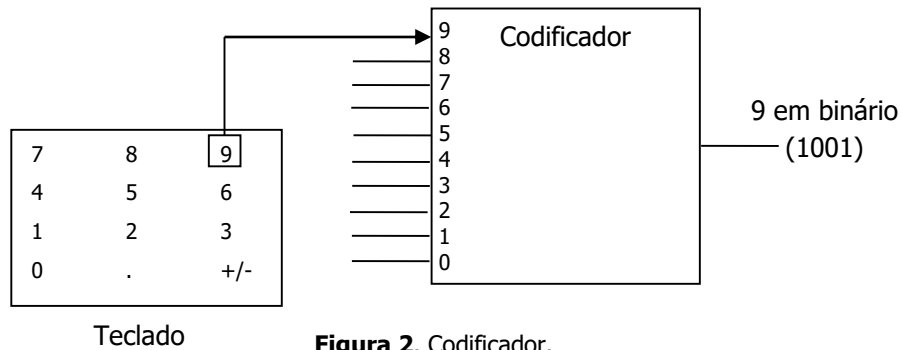


Figura 2. Codificador.

Fonte: THOMAS, L. F. *Sistemas digitais: fundamentos e aplicações*, 2007.

- Decodificação: converte uma informação codificada em alguma forma não codificada como, por exemplo, a conversão de um número binário em um decimal.

A figura 3 ilustra um decodificador utilizado para ativar um *display* de sete segmentos. Cada um dos sete segmentos do display é conectado a uma saída do decodificador. Quando determinado código binário aparece na entrada do decodificador, as saídas correspondentes são ativadas, fazendo com que os segmentos sejam acesos e mostrando o dígito decimal correspondente ao código binário.

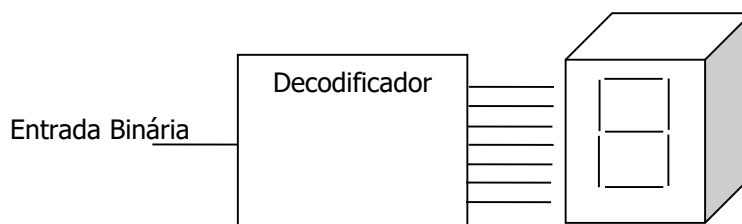


Figura 3. Decodificador.

Fonte: THOMAS, L. F. *Sistemas digitais: fundamentos e aplicações*, 2007.

1.3. Portas lógicas

As portas lógicas representam os elementos construtivos dos sistemas digitais. As portas lógicas e suas respectivas tabelas-verdade estão ilustradas a seguir.

AND

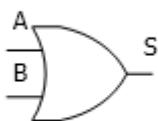


$$S = A.B$$

Tabela-Verdade

A	B	S
0	0	0
0	1	0
1	0	0
1	1	1

OR

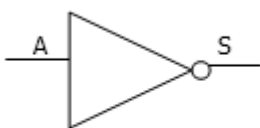


$$S = A + B$$

Tabela-Verdade

A	B	S
0	0	0
0	1	1
1	0	1
1	1	1

NOT



$$S = \overline{A}$$

Tabela-Verdade

A	S
0	1
1	0

XOR



$$S = A \oplus B$$

Tabela-Verdade

A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

NAND

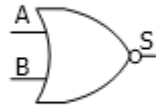


$$S = \overline{A.B}$$

Tabela-Verdade

A	B	S
0	0	1
0	1	1
1	0	1
1	1	0

NOR

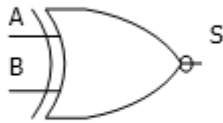


$$S = \overline{A + B}$$

Tabela-Verdade

A	B	S
0	0	1
0	1	0
1	0	0
1	1	0

XNOR



$$S = \overline{A \oplus B}$$

Tabela-Verdade

A	B	S
0	0	1
0	1	0
1	0	0
1	1	1

2. Indicação bibliográfica

- THOMAS, L. F. *Sistemas digitais: fundamentos e aplicações*. 9. ed. Porto Alegre: Bookman, 2007.

3. Análise das afirmativas

I – Afirmativa correta.

JUSTIFICATIVA. O circuito do item I tem as entradas S1 e S0 e as saídas 1 e 2. A partir das saídas da tabela e da porta lógica é possível montar a tabela verdade do circuito. A partir das entradas da tabela e da função lógica é possível montar a tabela verdade. Pela comparação das tabelas abaixo, concluímos que a função lógica corresponde ao circuito lógico (porta lógica OR). Veja a representação a seguir.

Tabela Verdade: circuito

1	2	circuito
0	0	0
1	0	1
0	1	1
0	0	0

Tabela Verdade: $f(A,B) = \bar{A}.B + A.\bar{B}$

A (S1)	B (S0)	\bar{A}	\bar{B}	$\bar{A}.B$	$A.\bar{B}$	$\bar{A}.B + A.\bar{B}$
0	0	1	1	0	0	0
0	1	1	0	1	0	1
1	0	0	1	0	1	1
1	1	0	0	0	0	0

II – Afirmativa correta.

JUSTIFICATIVA. O circuito do item II tem as entradas S1 e S0 e as saídas 0, 1 e 2. Assim como no item I, pela comparação das tabelas verdade apresentadas a seguir concluímos que a função lógica corresponde ao circuito lógico (porta lógica NOR). Veja a representação a seguir.

Tabela Verdade: circuito

0	1	2	circuito
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Tabela Verdade: $f(A,B) = A.B$

A (S1)	B (S2)	A.B
0	0	0
0	1	0
1	0	0
1	1	1

II – Afirmativa correta.

JUSTIFICATIVA. O circuito do item III tem as entradas S1 e S0 e as saídas 1, 2 e 3. Assim como nos itens I e II, pela comparação das tabelas abaixo concluímos que a função lógica corresponde ao circuito lógico (porta lógica XOR). Veja a representação a seguir.

Tabela Verdade: circuito

1	2	3	circuito
0	0	0	0
1	0	0	1
0	1	0	1
0	0	1	1

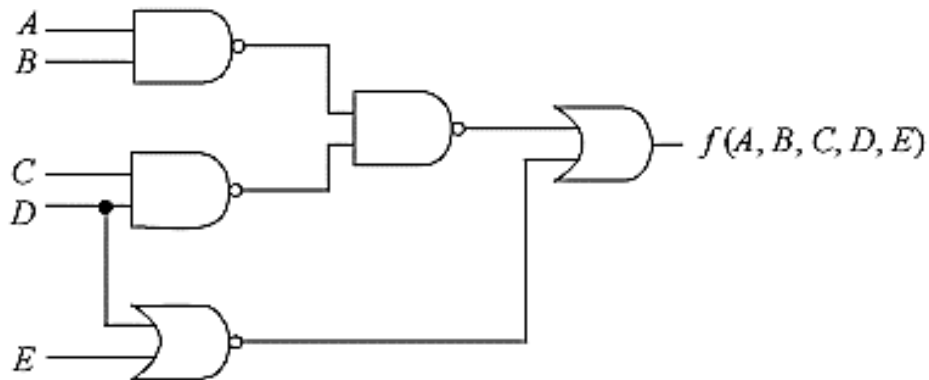
Tabela Verdade: $f(A,B) = A+B$

A (S1)	B (S2)	A + B
0	0	0
0	1	1
1	0	1
1	1	1

Alternativa correta: E.

Questão 5

Questão 5.⁵



No circuito acima, que possui cinco entradas — A, B, C, D e E — e uma saída $f(A, B, C, D, E)$, qual opção apresenta uma expressão lógica equivalente à função $f(A, B, C, D, E)$?

- Ⓐ $\overline{A.B} + \overline{C.D} + D.E$
- Ⓑ $(A+B).(C+D) + D.E$
- Ⓒ $\overline{A.B} + \overline{C.D} + D + E$
- Ⓓ $A.B + C.D + D + E$
- Ⓔ $A.B + C.D + \overline{D.E}$

1. Introdução teórica

Exemplo de circuito composto por três entradas

Considere o circuito ilustrado na figura 1, composto por três entradas.

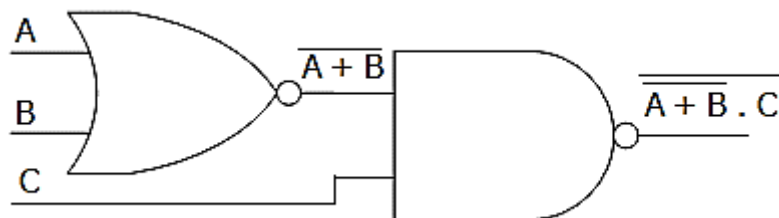


Figura 1. Exemplo de circuito composto por três entradas.

Como o circuito apresenta três entradas, a tabela verdade correspondente tem $2^3 = 8$ linhas (tabela 1).

⁵Questão 38 - Enade 2008.

Tabela 1. Tabela verdade referente ao circuito exemplo.

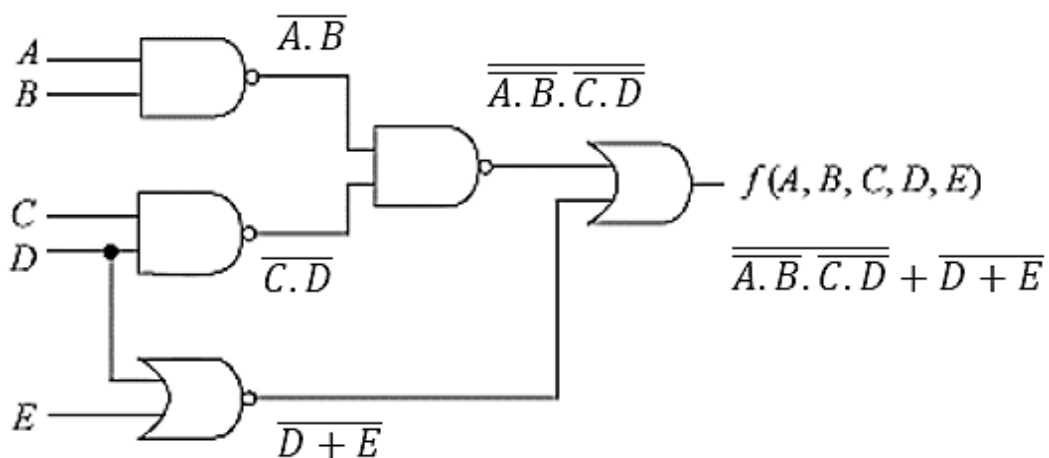
A	B	C	A + B	$\overline{A + B}$	$\overline{A + B \cdot C}$	$\overline{\overline{A + B \cdot C}}$
0	0	0	0	1	0	1
0	0	1	0	1	1	0
0	1	0	1	0	0	1
0	1	1	1	0	0	1
1	0	0	1	0	0	1
1	0	1	1	0	0	1
1	1	0	1	0	0	1
1	1	1	1	0	0	1

2. Indicações bibliográficas

- THOMAS, L. F. *Sistemas digitais: fundamentos e aplicações*. 9. ed. Porto Alegre: Bookman, 2007.

3. Análise das alternativas

O primeiro passo para resolvermos a questão é transformar o circuito do enunciado em uma expressão (figura 2).

**Figura 2.** Expressões do circuito do enunciado.

O circuito é formado por cinco entradas. Portanto, a tabela verdade correspondente (tabela 2) tem $2^5 = 32$ linhas.

Tabela 2. Tabela verdade do circuito do enunciado.

A	B	C	D	E	\overline{D}	\overline{E}	$\overline{A.B . C.D}$	$\overline{A.B . C.D+D+E}$
0	0	0	0	0	1	1	0	1
0	0	0	0	1	1	0	0	0
0	0	0	1	0	0	1	0	0
0	0	0	1	1	0	0	0	0
0	0	1	0	0	1	1	0	1
0	0	1	0	1	1	0	0	0
0	0	1	1	0	0	1	1	1
0	0	1	1	1	0	0	1	1
0	1	0	0	0	1	1	0	1
0	1	0	0	1	1	0	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	1	0	0	0	0
0	1	1	0	0	1	1	0	1
0	1	1	0	1	1	0	0	0
0	1	1	1	0	0	1	1	1
0	1	1	1	1	0	0	1	1
1	0	0	0	0	1	1	0	1
1	0	0	0	1	1	0	0	0
1	0	0	1	0	0	1	0	0
1	0	0	1	1	0	0	0	0
1	0	1	0	0	1	1	1	1
1	0	1	0	1	1	0	0	0
1	0	1	1	0	0	1	1	1
1	0	1	1	1	0	0	1	1
1	1	0	0	0	1	1	1	1
1	1	0	0	1	1	0	0	0
1	1	0	1	0	0	1	0	0
1	1	0	1	1	0	0	0	0
1	1	1	0	0	1	1	1	1
1	1	1	0	1	1	0	0	0
1	1	1	1	0	0	1	1	1
1	1	1	1	1	0	0	1	1

A – Alternativa incorreta.

JUSTIFICATIVA. A tabela verdade relativa à alternativa A (tabela 3) é diferente da tabela verdade do circuito (tabela 2).

Tabela 3. Tabela verdade relativa à alternativa A.

A	B	C	D	E	\overline{A}	\overline{B}	\overline{C}	\overline{D}	\overline{E}	$\overline{A.B}$	$\overline{C.D}$	D.E	$\overline{A.B} + \overline{C.D}$	$\overline{A.B} + \overline{C.D} + D.E$
0	0	0	0	0	1	1	1	1	1	1	1	0	1	1
0	0	0	0	1	1	1	1	1	0	1	1	0	1	1
0	0	0	1	0	1	1	1	0	1	1	0	0	1	1
0	0	0	1	1	1	1	1	0	0	1	0	1	1	1
0	0	1	0	0	1	1	0	1	1	1	0	0	1	1
0	0	1	0	1	1	1	0	1	0	1	0	0	1	1
0	0	1	1	0	1	1	0	0	1	1	0	0	1	1
0	0	1	1	1	1	1	0	0	0	1	0	1	1	1
0	1	0	0	0	1	0	1	1	1	0	1	0	1	1
0	1	0	0	1	1	0	1	1	0	0	1	0	1	1
0	1	0	1	0	1	0	1	0	1	0	0	0	0	0
0	1	0	1	1	1	0	1	0	0	0	0	1	0	1
0	1	1	0	0	1	0	0	1	1	0	0	0	0	0
0	1	1	0	1	1	0	0	1	0	0	0	0	0	0
0	1	1	1	0	1	0	0	0	1	0	0	0	0	0
0	1	1	1	1	1	0	0	0	0	0	0	1	0	1
1	0	0	0	0	0	1	1	1	1	0	1	0	1	1
1	0	0	0	1	0	1	1	1	0	0	1	0	1	1
1	0	0	1	0	0	1	1	0	1	0	0	0	0	0
1	0	0	1	1	0	1	1	0	0	0	0	1	0	1
1	0	1	0	0	0	1	0	1	1	0	0	0	0	0
1	0	1	0	1	0	1	0	1	0	0	0	0	0	0
1	0	1	1	0	0	1	0	0	1	0	0	0	0	0
1	0	1	1	1	0	1	0	0	0	0	0	1	0	1
1	1	0	0	0	0	0	1	1	1	0	1	0	1	1
1	1	0	0	1	0	0	1	1	0	0	1	0	1	1
1	1	0	1	0	0	0	1	0	1	0	0	0	0	0
1	1	0	1	1	0	0	1	0	0	0	0	1	0	1
1	1	1	0	0	0	0	0	1	1	0	0	0	0	0
1	1	1	0	1	0	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	1	0	1

B – Alternativa incorreta.

JUSTIFICATIVA. A tabela verdade relativa à alternativa B (tabela 4) é diferente da tabela verdade do circuito (tabela 2).

Tabela 4. Tabela verdade relativa à alternativa B.

A	B	C	D	E	A + B	C + D	D.E	(A+B) . (C +D)	(A+B) . (C+D)+D.E
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	1	0	0	1	0	0	0
0	0	0	1	1	0	1	1	0	1
0	0	1	0	0	0	1	0	0	0
0	0	1	0	1	0	1	0	0	0
0	0	1	1	0	0	1	0	0	0
0	0	1	1	1	0	1	1	0	1
0	1	0	0	0	1	0	0	0	0
0	1	0	0	1	1	0	0	0	0
0	1	0	1	0	1	1	0	1	1
0	1	0	1	1	1	1	1	1	1
0	1	1	0	0	1	1	0	1	1
0	1	1	0	1	1	1	0	1	1
0	1	1	1	0	1	1	0	1	1
0	1	1	1	1	1	1	1	1	1
1	0	0	0	0	1	0	0	0	0
1	0	0	0	1	1	0	0	0	0
1	0	0	1	0	1	1	0	1	1
1	0	0	1	1	1	1	1	1	1
1	0	1	0	0	1	1	0	1	1
1	0	1	0	1	1	1	0	1	1
1	0	1	1	0	1	1	0	1	1
1	0	1	1	1	1	1	1	1	1
1	1	0	0	0	1	0	0	0	0
1	1	0	0	1	1	0	0	0	0
1	1	0	1	0	1	1	0	1	1
1	1	0	1	1	1	1	1	1	1
1	1	1	0	0	1	1	0	1	1
1	1	1	0	1	1	1	0	1	1
1	1	1	1	0	1	1	0	1	1
1	1	1	1	1	1	1	1	1	1

C – Alternativa incorreta.

JUSTIFICATIVA. A tabela verdade relativa à alternativa C (tabela 5) é diferente da tabela verdade do circuito (tabela 2).

Tabela 5. Tabela verdade relativa à alternativa C.

A	B	C	D	E	\overline{A}	\overline{B}	\overline{C}	\overline{D}	$\overline{A.B}$	$\overline{C.D}$	D + E	$\overline{A.B} + \overline{C.D}$	$\overline{A.B} + \overline{C.D} + D + E$
0	0	0	0	0	1	1	1	1	1	1	0	1	1
0	0	0	0	1	1	1	1	1	1	1	1	1	1
0	0	0	1	0	1	1	1	0	1	0	1	1	1
0	0	0	1	1	1	1	1	0	1	0	1	1	1
0	0	1	0	0	1	1	0	1	1	0	0	1	1
0	0	1	0	1	1	1	0	1	1	0	1	1	1
0	0	1	1	0	1	1	0	0	1	0	1	1	1
0	0	1	1	1	1	1	0	0	1	0	1	1	1
0	1	0	0	0	1	0	1	1	0	1	0	1	1
0	1	0	0	1	1	0	1	1	0	1	1	1	1
0	1	0	1	0	1	0	1	0	0	0	1	0	1
0	1	0	1	1	1	0	1	0	0	0	1	0	1
0	1	1	0	0	1	0	0	1	0	0	0	0	0
0	1	1	0	1	1	0	0	1	0	0	1	0	1
0	1	1	1	0	1	0	0	0	0	0	1	0	1
0	1	1	1	1	1	0	0	0	0	0	1	0	1
1	0	0	0	0	0	1	1	1	0	1	0	1	1
1	0	0	0	1	0	1	1	1	0	1	1	1	1
1	0	0	1	0	0	1	1	0	0	0	1	0	1
1	0	0	1	1	0	1	1	0	0	0	1	0	1
1	0	1	0	0	0	1	0	1	0	0	0	0	0
1	0	1	0	1	0	1	0	1	0	0	1	0	1
1	0	1	1	0	0	1	0	0	0	0	1	0	1
1	0	1	1	1	0	1	0	0	0	0	1	0	1
1	1	0	0	0	0	0	1	1	0	1	0	1	1
1	1	0	0	1	0	0	1	1	0	1	1	1	1
1	1	0	1	0	0	0	1	0	0	0	1	0	1
1	1	0	1	1	0	0	1	0	0	0	1	0	1
1	1	1	0	0	0	0	0	1	0	0	0	0	1
1	1	1	0	1	0	0	0	1	0	0	1	0	1
1	1	1	1	0	0	0	0	0	0	0	1	0	1
1	1	1	1	1	0	0	0	0	0	0	1	0	1

D – Alternativa incorreta.

JUSTIFICATIVA. A tabela verdade relativa à alternativa D (tabela 6) é diferente da tabela verdade do circuito (tabela 2).

Tabela 6. Tabela verdade relativa à alternativa D.

A	B	C	D	E	A.B	C.D	D + E	A.B + C.D	A.B + C.D+ D+E
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	1	0	1
0	0	0	1	0	0	0	1	0	1
0	0	0	1	1	0	0	1	0	1
0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	1	0	1
0	0	1	1	0	0	1	1	1	1
0	0	1	1	1	0	1	1	1	1
0	1	0	0	0	0	0	0	0	0
0	1	0	0	1	0	0	1	0	1
0	1	0	1	0	0	0	1	0	1
0	1	0	1	1	0	0	1	0	1
0	1	1	0	0	0	0	0	0	0
0	1	1	0	1	0	0	1	0	1
0	1	1	1	0	0	1	1	1	1
0	1	1	1	1	0	1	1	1	1
1	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	1	0	1
1	0	0	1	0	0	0	1	0	1
1	0	0	1	1	0	0	1	0	1
1	0	1	0	0	0	0	0	0	0
1	0	1	0	1	0	0	1	0	1
1	0	1	1	0	0	1	1	1	1
1	0	1	1	1	0	1	1	1	1
1	1	0	0	0	1	0	0	1	1
1	1	0	0	1	1	0	1	1	1
1	1	0	1	0	1	0	1	1	1
1	1	0	1	1	1	0	1	1	1
1	1	1	0	0	1	0	0	1	1
1	1	1	0	1	1	0	1	1	1
1	1	1	1	0	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

E – Alternativa correta.

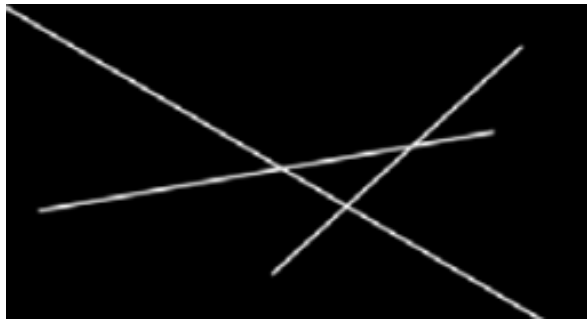
JUSTIFICATIVA. A tabela verdade relativa à alternativa E (tabela 7) é igual à tabela verdade do circuito (tabela 2).

Tabela 7. Tabela verdade relativa à alternativa E.

A	B	C	D	E	\overline{D}	\overline{E}	A.B	C.D	$\overline{D.E}$	A.B + C.D	A.B + C.D + $\overline{D.E}$
0	0	0	0	0	1	1	0	0	1	0	1
0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	1	0	0	1	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0
0	0	1	0	0	1	1	0	0	1	0	1
0	0	1	0	1	1	0	0	0	0	0	0
0	0	1	1	0	0	1	0	1	0	1	1
0	0	1	1	1	0	0	0	1	0	1	1
0	1	0	0	0	1	1	0	0	1	0	1
0	1	0	0	1	1	0	0	0	0	0	0
0	1	0	1	0	0	1	0	0	0	0	0
0	1	0	1	1	0	0	0	0	0	0	0
0	1	1	0	0	1	1	0	0	1	0	1
0	1	1	0	1	1	0	0	0	0	0	0
0	1	1	1	0	0	1	0	1	0	1	1
0	1	1	1	1	0	0	0	1	0	1	1
1	0	0	0	0	1	1	0	0	1	0	1
1	0	0	0	1	1	0	0	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0
1	0	0	1	1	0	0	0	0	0	0	0
1	0	1	0	0	1	1	0	0	1	1	1
1	0	1	0	1	1	0	0	0	0	0	0
1	0	1	1	0	0	1	0	1	0	1	1
1	0	1	1	1	0	0	0	1	0	1	1
1	1	0	0	0	1	1	1	0	1	1	1
1	1	0	0	1	1	0	1	0	0	0	0
1	1	0	1	0	0	1	1	0	0	0	0
1	1	0	1	1	0	0	1	0	0	0	0
1	1	1	0	0	1	1	1	0	1	1	1
1	1	1	0	1	1	0	1	0	0	0	0
1	1	1	1	0	0	1	1	0	1	1	1
1	1	1	1	1	0	0	1	1	0	1	1

Questão 6

Questão 6.⁶



A figura acima ilustra uma imagem binária com pixels brancos formando retas sobre um fundo preto. Com relação à aplicação de transformadas sobre essa imagem, assinale a opção correta.

- A. A transformada de Fourier, quando aplicada à imagem descrita, produz como resultado um mapa de frequências que equivale ao histograma dos níveis de cinza das retas presentes.
- B. A transformada de Hadamard da imagem apresentada tem resultado equivalente à aplicação de um filtro passa-baixas, o que destaca as retas existentes.
- C. Ao se aplicar a transformada da distância à imagem binária, considerando pixels brancos como objetos, são geradas as distâncias entre as retas presentes e o centro da imagem, o que permite identificar as equações das retas formadas na imagem.
- D. O uso da transformada dos cossenos produz uma lista dos coeficientes lineares e angulares das diversas retas existentes nessa imagem binária.
- E. O resultado da aplicação da transformada de Hough usando parametrização de retas é um mapa cujos picos indicam os pixels colineares, permitindo que sejam identificados coeficientes que descrevem as diversas retas formadas na imagem.

1. Introdução teórica

Processamento de imagens

A obtenção e o processamento de imagens envolvem as seguintes etapas: aquisição, armazenamento, processamento e exibição.

Uma imagem corresponde a uma matriz cujas linhas e colunas identificam seus pontos. Depois de captura, é preciso digitalizar a imagem para processá-la. A digitalização das

⁶Questão 25 - Enade 2008.

coordenadas espaciais (x,y) é a amostragem da imagem e a digitalização da amplitude é a quantização em níveis de cinza. Após a digitalização da imagem $f(x,y)$, gera-se a matriz na qual cada elemento é uma quantidade discreta.

A imagem pode ser considerada uma forma de sinal na qual se usam técnicas de análise e de processamento de sinais para, por exemplo, a aplicação de métodos para a compressão, a codificação, a filtragem, o realce e a suavização. Tais métodos são chamados de transformadas.

Uma das ferramentas matemáticas utilizadas na área do processamento de imagens é a transformada de Fourier, que representa a decomposição da imagem original em seus componentes de frequência espacial nas direções horizontal e vertical.

A transformada dos cossenos, uma ferramenta utilizada no processo de codificação, é um caso particular de uma classe mais geral de transformadas cujas matrizes de transformação têm suas linhas compostas por senos e cossenos de frequência crescente. Em aplicações envolvendo sinais de vídeo, é necessário um grande fator de compressão por parte do codificador.

A transformada de Hadamard tem núcleos bidimensionais e não é baseada em funções senoidais.

A transformada de Hough é um método padrão para a detecção de formas que podem ser representadas, em imagens digitalizadas, por equações parametrizadas (linhas, círculos, elipses etc.).

2. Indicação bibliográfica

- GONZALEZ, R. G.; WOODS, R. *Processamento digital de imagens*. São Paulo: Edgard Blücher, 2000.

3. Análise das alternativas

A – Alternativa incorreta.

JUSTIFICATIVA. A transformada de Fourier representa a decomposição da imagem original em seus componentes de frequência espacial nas direções horizontal e vertical. O histograma de uma imagem digital é um gráfico que mostra a quantidade de vezes que dada cor se repete, permitindo indicar o percentual de pixels que a imagem tem de determinado nível de cinza ou cor. A transformada de Fourier não é equivalente a um histograma dos níveis de cinza.

B – Alternativa incorreta.

JUSTIFICATIVA. Os filtros passa-baixa suavizam a imagem e atenuam as altas frequências que correspondem às transições abruptas. Eles minimizam ruídos e apresentam o efeito de borramento da imagem. A transformada de Hadamard não tem as características de um filtro passa-baixa.

C – Alternativa incorreta.

JUSTIFICATIVA. A transformada da distância à imagem binária aplicada a uma imagem gera outra imagem na qual o valor para cada pixel representa a menor distância entre esse pixel e outro objeto da imagem, não tendo relação com a distância entre as retas e o centro da imagem.

D – Alternativa incorreta.

JUSTIFICATIVA. A transformada dos cossenos tem suas linhas compostas de senos e cossenos de frequências crescentes, gerando uma lista dos coeficientes das retas existentes na imagem.

E – Alternativa correta.

JUSTIFICATIVA. A transformada de Hough é o método padrão para a detecção de formas, em imagens digitalizadas, por equações parametrizadas como linhas, círculos, elipses etc.

Questão 7

Questão 7.⁷



Figura I



Figura II

As figuras I e II apresentam duas imagens, ambas com resolução de 246 pixels \times 300 pixels, sendo que a figura I apresenta 256 níveis de cinza e a figura II, 4 níveis de cinza.

Considere que a imagem da figura I seja a original, tendo sido manipulada em um único atributo para gerar a imagem da figura II. Nessa situação, em qual atributo se diferenciam as imagens I e II acima?

- A. Resolução.
- B. Quantização.
- C. Iluminação.
- D. Escala.
- E. Amostragem espacial.

⁷Questão 26 - Enade 2008.

1. Introdução teórica

1.1. Propriedades da imagem

Uma imagem digital pode ser representada por uma matriz de valores inteiros não negativos na qual cada elemento corresponde a um pixel da imagem. As imagens digitais são funções bidimensionais discretas da forma $f(x,y)$, sendo f um valor inteiro não negativo que representa o nível de intensidade luminosa proveniente de uma cena registrada por um sensor óptico na coordenada espacial (x,y) .

Inicialmente, a imagem analógica de uma cena apresenta valores de f contínuos dentro de determinada faixa finita e proporcionais à intensidade luminosa incidente no sensor. Ou seja, $0 \leq f(x, y) < \infty$.

A função f pode ser interpretada como uma combinação dos seguintes dois componentes:

- a quantidade de luz proveniente da fonte luminosa da cena, ou seja, a iluminação $i(x,y)$, que depende exclusivamente da fonte luminosa.
- a quantidade de luz refletida pelos objetos da cena, ou seja, a reflectância $r(x,y)$, que depende das características superficiais dos objetos da cena, variando desde a total absorção da luz ($r=0$) até a total reflexão dos raios luminosos ($r=1$).

Após ser captada pelo sensor óptico, a imagem passa por um processo de digitalização ou discretização do sinal, destacando-se duas tarefas fundamentais realizadas pelo conversor analógico-digital: a amostragem e a quantização do sinal.

A amostragem, ou amostragem espacial, consiste em medir o valor da função $f(x,y)$ em intervalos discretos no espaço. Está relacionada com o número de pontos amostrados de uma imagem digitalizada (resolução).

A quantização, ou quantificação, é o processo de substituir a variação contínua de $f(x,y)$ por um conjunto discreto de valores como, por exemplo, o valor dos níveis de tons de cinza que pode ser atribuído a cada ponto digitalizado.

A amostragem e a quantização de uma imagem estão intimamente ligadas à qualidade da imagem digital resultante. Quanto maior o número de pontos amostrados na imagem, maior sua resolução espacial. Quanto maior o número de níveis de intensidade luminosa considerados na imagem para representar a tonalidade de um ponto amostrado, maior sua resolução de tonalidades.

1.2. Amostragem e quantização

A amostragem refere-se à divisão do plano xOy em uma grade, sendo x e y números inteiros não negativos que representam, respectivamente, a largura e a altura em pixels. Vejamos um exemplo na figura 1.

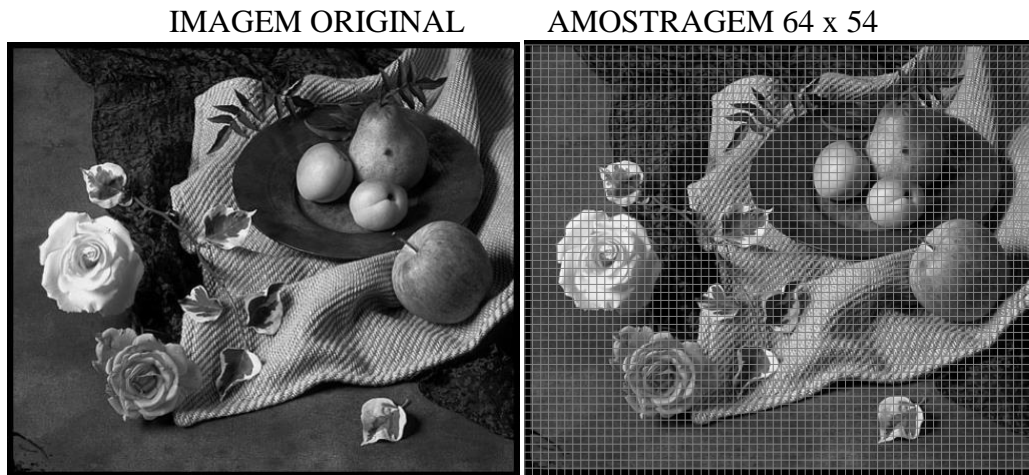


Figura 1. Exemplo de amostragem.

A quantificação, ou quantização, refere-se à representação discreta da intensidade luminosa (nível de cinza ou *gray level*) de cada pixel da amostra. Nas figuras 2, 3 e 4, há exemplos de visualizações quantificadas em 1D, 2D e 3D, respectivamente.

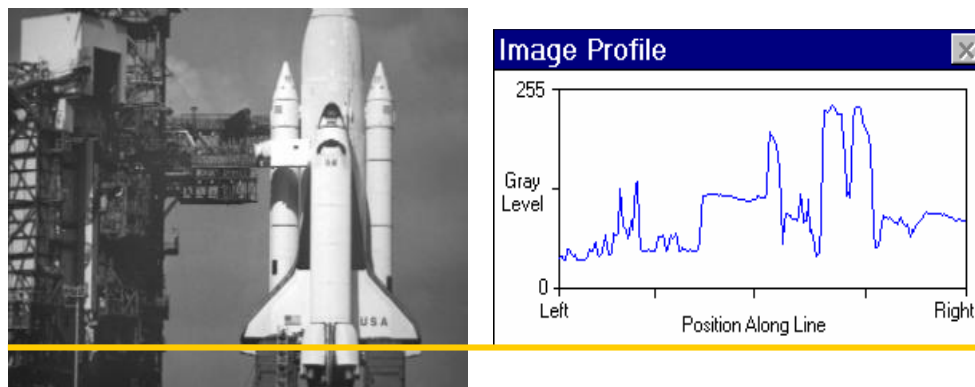


Figura 2. Exemplo em 1D de visualização quantificada.

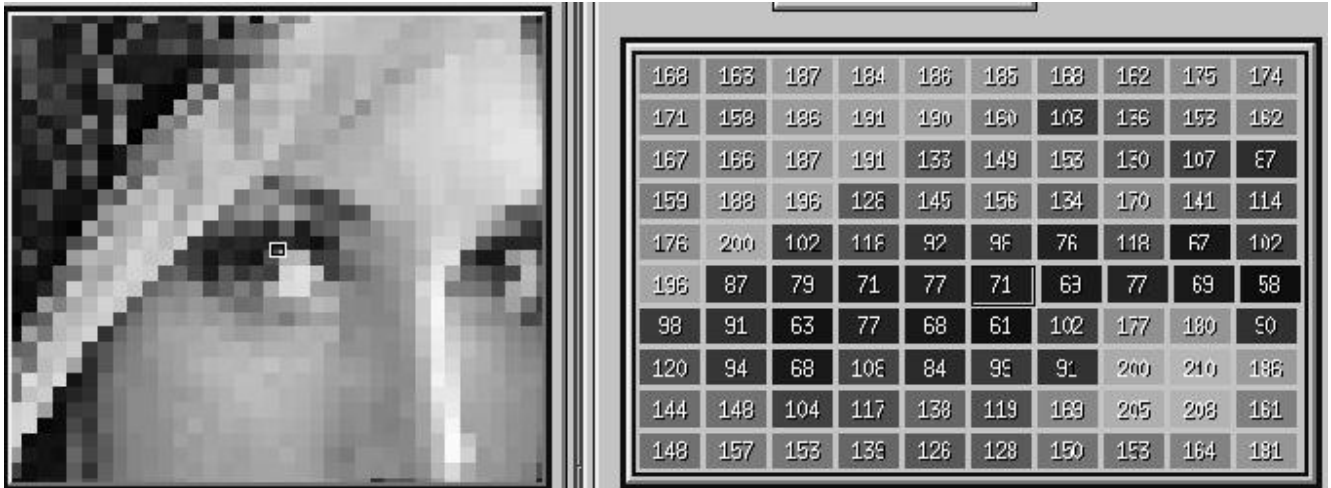


Figura 3. Exemplo em 2D de visualização quantificada.

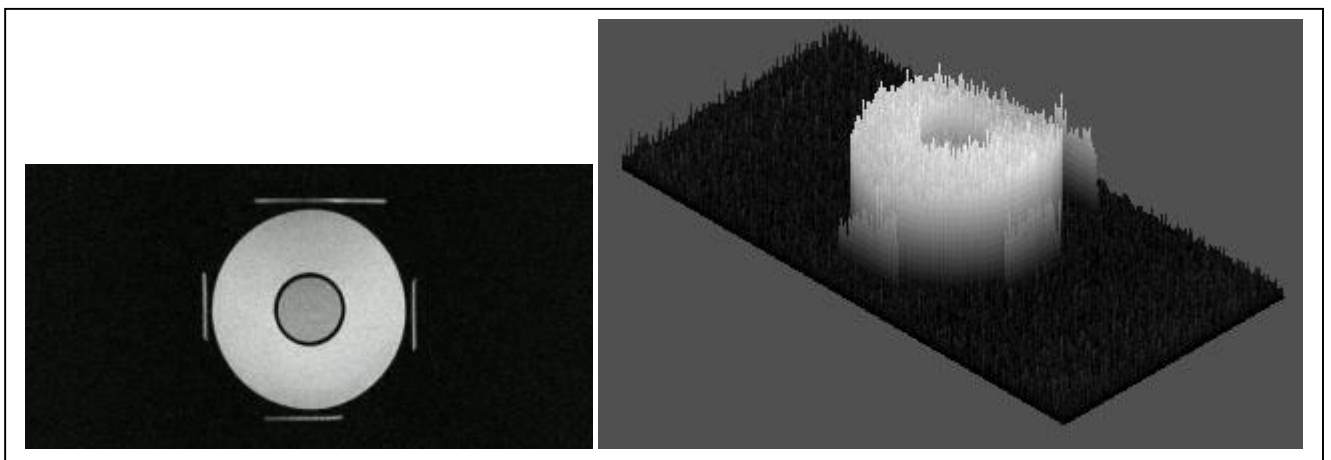


Figura 4. Exemplo em 3D de visualização quantificada.

2. Indicações bibliográficas

- CONCI, A.; AZEVEDO, E.; LETA, F. R. *Computação gráfica - teoria e prática*. V. 2. Rio de Janeiro: Campus, 2008.
- GONZALEZ, R. G.; WOODS, R. *Processamento digital de imagens*. 3. ed. São Paulo: Pearson, 2010.
- PEDRINI, H.; SCHWARTZ, W. R. *Análise de imagens digitais*. São Paulo: Thomson, 2008.

3. Análise das alternativas

A – Alternativa incorreta.

JUSTIFICATIVA. A resolução é a quantidade de pixels de uma imagem nas direções x e y. No enunciado da questão é dito que as figuras I e II apresentam a mesma resolução de 246 x 300 pixels. Portanto, no que se refere à resolução, não há diferença entre as duas imagens apresentadas.

B – Alternativa correta.

JUSTIFICATIVA. A quantização refere-se ao valor, ou nível de tons de cinza, que pode ser atribuído a cada ponto ou pixel. Cada pixel de uma imagem digital pode ter um valor inteiro não negativo que corresponde à quantificação estabelecida. No enunciado da questão é dito que a figura I apresenta 256 níveis de cinza (quantificada com 256 níveis de cinza), o que significa que cada pixel pode assumir qualquer valor entre 0 e 255 ($L = 256$ e $L-1 = 255$). Já a figura II sofreu uma requantização para 4 níveis de cinza, ou seja, cada pixel só pode assumir valores entre 0 e 3. Sendo assim, o atributo que diferencia a imagem I da imagem II é a quantização.

C – Alternativa incorreta.

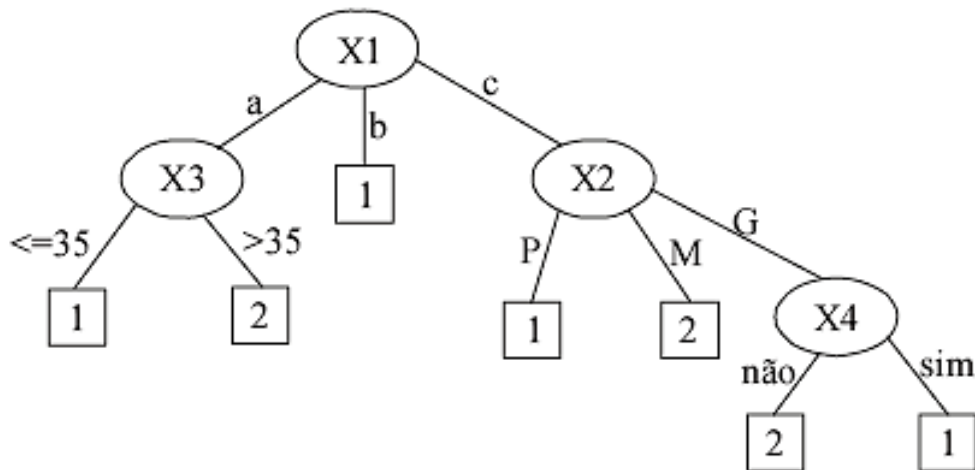
JUSTIFICATIVA. Iluminação é a quantidade de luz proveniente da fonte luminosa que ilumina a cena capturada pelo sensor óptico para, então, ser digitalizada pelos processos de amostragem e quantização. Logo, a iluminação sequer é um atributo da imagem, que depende ainda da reflectância dos objetos em cena para compor a intensidade luminosa que será registrada pelo sensor.

D – Alternativa incorreta.

JUSTIFICATIVA. Trata-se de uma transformação geométrica no espaço. No contexto da questão, ela pode confundir-se com a resolução. Ou seja, se a figura II passasse a ter resolução de 123×150 pixels, poderíamos argumentar que ela sofreu diminuição de tamanho de um fator de escala de aproximadamente 0,5 em ambas as dimensões. Mas não foi o caso, já que ambas as imagem permaneceram com o mesmo tamanho.

E – Alternativa incorreta.

JUSTIFICATIVA. A amostragem espacial é a discretização da imagem capturada pelo sensor óptico em um número definido de pixels nas direções espaciais x e y. Há outro tipo de amostragem, a temporal, mas ela diz respeito, por exemplo, ao sinal sonoro, à resolução temporal de quadros em uma animação etc. De qualquer modo, a alternativa E é incorreta pelos mesmos argumentos utilizados para refutar a alternativa A. Ou seja, as imagens I e II apresentam a mesma resolução e, portanto, os mesmos atributos de amostragem.

Questão 8**Questão 8.⁸**

A figura acima mostra uma árvore de decisão construída por um algoritmo de aprendizado indutivo a partir de um conjunto de dados em que os objetos são descritos por 4 atributos: X1, X2, X3 e X4. Dado um objeto de classe desconhecida, essa árvore classifica o objeto na classe 1 ou na classe 2. A tabela a seguir apresenta três objetos a serem classificados: O1, O2 e O3.

Objeto	X1	X2	X3	X4
O1	a	P	20	não
O2	b	M	21	não
O3	c	M	10	sim

A que classes corresponderiam, respectivamente, os objetos O1, O2 e O3?

- A. 1, 1 e 2. B. 1, 2 e 1. C. 2, 1 e 2. D. 2, 2 e 1. E. 1, 1 e 1.

1. Introdução teórica**Árvores de decisão**

As árvores de decisão são estruturas em sistemas especialistas e em problemas de classificação, sendo que as entradas são situações descritas por um conjunto de atributos e o retorno é uma decisão. A decisão é obtida após uma sequência de testes.

⁸Questão 28 - Enade 2008.

Cada nó interno da árvore corresponde a um teste do valor de uma das propriedades e os ramos do nó são identificados com os possíveis valores do teste. Cada nó folha da árvore especifica o valor de retorno se a folha for atingida.

2. Indicação bibliográfica

- STUART R.; PETER N. *Artificial intelligence, a modern approach*. 2. ed. New York: Prentice Hall, 2003.

3. Análise das alternativas

A – Alternativa correta.

JUSTIFICATIVA. Para o objeto O1, o caminho percorrido a partir do nó X1 segue pela aresta a e prossegue para o nó X3. Como a propriedade tem o valor 20, o caminho será em direção à folha 1. Como os nós folhas representam as classes, a classe correspondente ao objeto O1 é a 1. O caminho correspondente ao nó X2 pode ser ignorado. Para o objeto O2, o caminho percorrido a partir do X1 segue pela aresta b, caminhando para o nó folha 1, que também indica pertencer à classe 1. Para o objeto O3, o caminho percorrido a partir do nó X1 segue pela aresta c e prossegue para o nó X2, onde a propriedade assume o valor M, e vai para o nó folha 2, o que indica que a classe do objeto O3 é a 2.

B – Alternativa incorreta.

JUSTIFICATIVA. Para que o objeto O2 pertencesse à classe 2, o caminho deveria ter como propriedade em X1 as arestas a ou c. X3 deveria ser maior do que 35 ou X2 deveria ter a propriedade M ou G, seguindo para X4, que deveria ter a propriedade não. Para que o objeto O3 fosse da classe 1, a propriedade de X2 deveria ser 1.

C – Alternativa incorreta.

JUSTIFICATIVA. Para que o objeto O1 fosse da classe 2, a propriedade relativa ao nó X3 deveria ser maior do que 35.

D – Alternativa incorreta.

JUSTIFICATIVA. Para que o objeto O1 fosse da classe 2, a propriedade relativa ao nó X3 deveria ser maior do que 35. Para o objeto O2, a propriedade de X1 deveria ser a ou c, com

a propriedade maior do que 35 para X3 ou a propriedade M ou G para X2. Se o valor da propriedade fosse G, o nó X4 deveria ter propriedade não. Para o objeto O3, a propriedade de X2 deveria ser igual a P ou G, com a propriedade do nó X4 valendo sim.

E – Alternativa incorreta.

JUSTIFICATIVA. Para que o objeto O3 pertencesse à classe 1, o nó X2 deveria ter a propriedade P ou G e X4 deveria ter a propriedade sim.

Questão 9**Questão 9.⁹**

Julgue os itens a seguir, relativos a métodos de busca com informação (busca heurística) e sem informação (busca cega), aplicados a problemas em que todas as ações têm o mesmo custo, o grafo de busca tem fator de ramificação finito e as ações não retornam a estados já visitados.

- I. A primeira solução encontrada pela estratégia de busca em largura é a solução ótima.
- II. A primeira solução encontrada pela estratégia de busca em profundidade é a solução ótima.
- III. As estratégias de busca com informação usam funções heurísticas que, quando bem definidas, permitem melhorar a eficiência da busca.
- IV. A estratégia de busca gulosa é eficiente porque expande apenas os nós que estão no caminho da solução.

São corretos apenas os itens

- A. I e II. B. I e III. C. I e IV. D. II e IV. E. III e IV.

1. Introdução teórica**1.1. Resolução de problemas**

Um problema pode ser visto como uma tripla $\{I, O, B\}$, sendo que I representa os estados iniciais, O, o conjunto de operações e B, o estado objetivo. Uma solução para um problema é uma sequência finita de operações que permite sair de um elemento no estado inicial I e chegar a um elemento no estado B.

Um sistema de resolução de problemas suporta os seguintes elementos:

- um conjunto de estruturas de dados organizado em um grafo;
- um conjunto de operadores caracterizados por suas condições de aplicação e sua ação;
- uma estrutura de controle implementando a estratégia de resolução.

As abordagens de busca básicas em um espaço de estados são:

- busca cega (sem informação/não informada), ou seja, não há informação sobre qual sucessor é mais promissor para atingir a meta;

⁹Questão 31 - Enade 2008.

- busca heurística (com informação/informada), ou seja, há informação (estimativa) sobre qual sucessor é mais promissor para atingir a meta.

As estratégias de buscas cegas são: busca em largura, busca de custo uniforme, busca em profundidade, busca em profundidade limitada, busca em profundidade com aprofundamento iterativo, busca bidirecional, busca evitando estados repetidos e busca com conhecimento incompleto

As estratégias de busca distinguem-se pela ordem em que os nós são expandidos.

1.2. Busca em profundidade

A busca é iniciada na raiz e avança em níveis cada vez mais profundos. Um operador é aplicado a um nó para gerar, na sequência, um nó mais profundo. O processo continua até que uma solução seja encontrada ou um retrocesso seja forçado ao atingir-se um nó terminal que não é solução.

Vejamos um exemplo: o caminho para se chegar ao nó G da figura 1, usando busca em profundidade, é { B, F, C, E, J, K, D, G }.

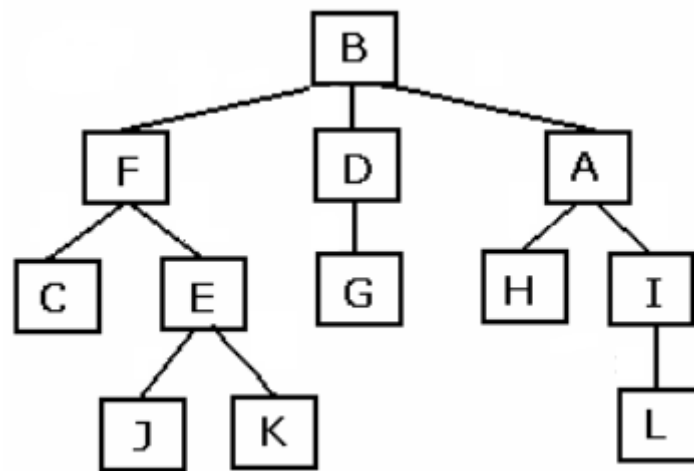


Figura 1. Busca em profundidade.

Fonte. RUSSELL, S. J.; NORVIG, P. *Inteligência Artificial*. São Paulo: Campus, 2004.

1.3. Busca em largura

Os nós em cada nível da árvore são completamente examinados antes haver movimento para o nível seguinte. Uma busca em largura sempre encontrará o menor caminho entre o estado inicial e o estado-objetivo. O menor caminho é o que apresenta menor número de passos, não devendo ser confundido com o caminho de menor custo. Ele faz uma busca sistemática, examinando primeiramente os módulos próximos à raiz.

Os critérios na análise de um algoritmo de busca incluem:

- completeza (o algoritmo oferece a garantia de encontrar uma solução quando ela existir?);
- otimização (a estratégia encontra a solução ótima, de menor custo de caminho, entre todas as soluções?);
- complexidade de tempo (quanto tempo leva-se para encontrar uma solução?);
- complexidade de espaço (quanto de memória é necessário para executar a busca?).

1.4. Estratégias de buscas heurísticas

As buscas em largura e em profundidade não fazem uso de nenhum conhecimento para encontrar sua solução, realizando uma busca exaustiva dentro do seu espaço. Para contornarmos esse problema, podemos utilizar métodos heurísticos.

A heurística corresponde à informação específica do domínio que pode ser usada para guiar o processo de busca. Em muitos casos, uma heurística envolve a aplicação de uma função que avalia um nó particular e prediz a qualidade dos seus nós sucessores.

Os tipos de buscas heurísticas são:

- busca do menor caminho - pode ser derivada de um refinamento da busca em largura;
- busca em largura - sempre são escolhidos, para expansão, os menores caminhos-candidatos (isto é, os nós extremos menos profundos da busca);
- busca do melhor caminho – calcula-se uma estimativa heurística para cada candidato e escolhe-se, para expansão, o melhor candidato de acordo com essa estimativa;
- busca gulosa – tenta-se expandir o nó mais próximo à meta, supondo-se que isso provavelmente levará a uma solução rápida.

2. Indicação bibliográfica

- RUSSELL, S. J.; NORVIG, P. *Inteligência artificial*. São Paulo: Campus, 2004.

3. Análise das afirmativas

I – Afirmativa correta.

JUSTIFICATIVA. Os nós em cada nível da árvore são completamente examinados antes de haver movimentação para o próximo nível, ou seja, antes de testar um novo nodo em um nível n , todos os nodos do nível $n-1$ são verificados. Uma busca em largura sempre encontrará

o menor caminho entre o estado inicial e o estado objetivo. Como todas as ações têm o mesmo custo, se uma solução for encontrada em um nível n , ela será melhor do que uma solução de um nível $m > n$. Como todos os nodos de um mesmo nível têm o mesmo custo, a solução pode ser considerada ótima.

II – Afirmativa incorreta.

JUSTIFICATIVA. A busca é iniciada na raiz e avança em níveis cada vez mais profundos. Um operador é aplicado a um nó para gerar, na sequência, um nó mais profundo. O processo continua até que uma solução seja encontrada ou um retrocesso seja forçado ao atingir-se um nó terminal que não é solução. A busca em profundidade prioriza a expansão dos filhos de cada nodo. Como o custo da solução é dado pelo seu valor constante c multiplicado pelo nível do nodo, essa solução pode desprezar uma solução em um nível anterior, o que permite a utilização de uma solução não ótima.

III – Afirmativa correta.

JUSTIFICATIVA. Em muitos casos, uma heurística envolve a aplicação de uma função que avalia um nó particular e prediz a qualidade dos seus nós sucessores. Uma heurística bem elaborada pode reduzir o número de nós que devem ser testados, tornando a busca mais produtiva.

IV – Afirmativa incorreta.

JUSTIFICATIVA. Tenta-se expandir o nó mais próximo à meta, na suposição de que isso provavelmente levará a uma solução rápida. Avaliam-se nós para expansão com base unicamente na função heurística $f(n) = h(n)$. A busca gulosa não garante que o caminho escolhido realize o melhor percurso para a solução do problema.

Alternativa correta: B.

ÍNDICE REMISSIVO

Questão 1	Índices. Índices ordenados. Estruturas de dados utilizados na implementação de índices.
Questão 2	Programas de computadores. Processador de linguagem.
Questão 3	Teoria relacional. Regras de integridade (<i>constraints</i>).
Questão 4	Operações lógicas básicas. Funções lógicas básicas. Portas lógicas.
Questão 5	Exemplo de circuito composto por três entradas.
Questão 6	Processamento de imagens.
Questão 7	Propriedades da imagem. Amostragem e quantização.
Questão 8	Árvores de decisão.
Questão 9	Resolução de problemas. Busca em profundidade. Busca em largura. Estratégias de buscas heurísticas.