

Object Pascal

Ementa

- ▶ Instalação e Configuração do Ambiente de Desenvolvimento
- ▶ Sintaxe da Linguagem
- ▶ Operadores
- ▶ Tipos de Dados
- ▶ Estruturas de Seleção
- ▶ Estruturas de Repetição
- ▶ Debugger
- ▶ Procedures e Functions
- ▶ Arrays e Records
- ▶ +30 Exercícios Hands On

Revisão

Carango Velho

- ▶ 27. A concessionária de veículos “CARANGO” está vendendo os seus veículos com desconto. Faça um algoritmo que calcule e exiba o valor do desconto e o valor a ser pago pelo cliente. O desconto deverá ser calculado sobre o valor do veículo de acordo com o combustível (álcool - 25%, gasolina - 21% ou diesel -14%). Com valor do veículo zero encerra entrada de dados. total de desconto e total pago pelos clientes;

Trabalhando com Menus

Certificação

Question 1 of 60

Fundamentals

The following is a valid constant declaration: `const Zero = 0;`

Correct answer: A.

You chose: A.

A. True

B. False

Question 1 of 60

Fundamentals

The following is a valid constant declaration: `const Zero = 0;`

Correct answer: A.

You chose: A.

✓ A. True

B. False

Question 2 of 60

Fundamentals

How many uses clauses can a Unit contain?

Correct answer: D.

You chose: D.

This question displayed answers in random order during the test.

- ☒ A. Any number of uses clauses
 - ☐ B. One or two uses clauses
 - ☐ C. Exactly one uses clause
 - ☐ D. Zero, one, or two uses clauses
-

Question 2 of 60

Fundamentals

How many uses clauses can a Unit contain?

Correct answer: D.

You chose: D.

This question displayed answers in random order during the test.

- A. Any number of uses clauses
 - B. One or two uses clauses
 - C. Exactly one uses clause
 - ✓ D. Zero, one, or two uses clauses
-

Question 3 of 60

Fundamentals

The `>` (greater than) operator has a higher precedence than the negation (not) operator.

Correct answer: B.

You chose: B.

A. True

B. False

Question 3 of 60

Fundamentals

The > (greater than) operator has a higher precedence than the negation (not) operator.

Correct answer: B.

You chose: B.

A. True

✓ B. False

PRECEDÊNCIA DE OPERADORES

Precedência	Operador
Primeira	@, not
Segunda	*, /, div, mod, and, shl, shr, as
Terceira	+, -, or, xor
Quarta	=, <>, >=, <=, >, <

Question 4 of 60

Fundamentals

In which source file is the first line most likely to contain the reserved word "program"?

Correct answer: C.

You chose: C.

This question displayed answers in random order during the test.

- A. The DCU file
 - B. The DFM file
 - C. The DPR file
 - D. The PAS file
-

Question 4 of 60

Fundamentals

In which source file is the first line most likely to contain the reserved word "program"?

Correct answer: C.

You chose: C.

This question displayed answers in random order during the test.

- A. The DCU file
- B. The DFM file
- ✓ C. The DPR file
- D. The PAS file

Question 5 of 60

Fundamentals

In the Delphi language, which symbols or words define the start and end of a code block?

Correct answer: D.

You chose: D.

This question displayed answers in random order during the test.

A. []

B. {}

C. Start End

D. Begin End

Question 5 of 60

Fundamentals

In the Delphi language, which symbols or words define the start and end of a code block?

Correct answer: D.

You chose: D.

This question displayed answers in random order during the test.

- A. []
 - B. {}
 - C. Start End
 - ✓ D. Begin End
-

Question 6 of 60

Fundamentals

Assuming that `UpperCase` is a function declared in the `SysUtils` unit, it is possible to invoke `UpperCase` by qualifying it by its unit name, as shown here:

```
if  
SysUtils.UpperCase(SomeValue) = 'A' then ...
```

Correct answer: A.

You chose: A.

☒ A. True

☐ B. False

Question 6 of 60

Fundamentals

Assuming that `UpperCase` is a function declared in the `SysUtils` unit, it is possible to invoke `UpperCase` by qualifying it by its unit name, as shown here:

```
if  
SysUtils.UpperCase(SomeValue) = 'A' then ...
```

Correct answer: **A.**

You chose: **A.**

✓ **A. True**

B. False

Question 7 of 60

Fundamentals

The DFM file can be edited in the Delphi Code Editor.

Correct answer: A.

You chose: A.

A. True

B. False

Question 7 of 60

Fundamentals

The DFM file can be edited in the Delphi Code Editor.

Correct answer: A.

You chose: A.

✓ A. True

B. False

Question 8 of 60

Data types, variables, & Const

The following record declaration is correct.

```
TDateRecord = record  
  Year: 1990..2011;  
  Month: (mtJan, mtFeb, mtMar, mtApr, mtMay, mtJun, mtJul, mtAug, mtSep, mtOct, mtNov,  
mtDec);  
  Day: 1..31;  
end;
```

Correct answer: **A.**

You chose: **A.**

A. True

B. False

Question 8 of 60

Data types, variables, & Const

The following record declaration is correct.

```
TDateRecord = record  
  Year: 1990..2011;  
  Month: (mtJan, mtFeb, mtMar, mtApr, mtMay, mtJun, mtJul, mtAug, mtSep, mtOct, mtNov,  
  mtDec);  
  Day: 1..31;  
end;
```

Correct answer: A.

You chose: A.

✓ A. True

B. False

Question 17 of 60

Procedures and Functions

Which of the following procedures produces a compiler error?

Correct answer: C.

You chose: C.

This question displayed answers in random order during the test.

- A.

```
procedure Test(X: Integer);  
begin  
  X := 42;  
end;
```
- B.

```
procedure Test(out X: Integer);  
begin  
  X := 42;  
end;
```
- C.

```
procedure Test(const X: Integer);  
begin  
  X := 42;  
end;
```
- D.

```
procedure Test(var X: Integer);  
begin  
  X := 42;  
end;
```


Question 17 of 60

Procedures and Functions

Which of the following procedures produces a compiler error?

Correct answer: C.

You chose: C.

This question displayed answers in random order during the test.

A. `procedure Test(X: Integer);
begin
 X := 42;
end;`

B. `procedure Test(out X: Integer);
begin
 X := 42;
end;`

✓ C. `procedure Test(const X: Integer);
begin
 X := 42;
end;`

D. `procedure Test(var X: Integer);
begin
 X := 42;
end;`

Question 18 of 60

Procedures and Functions

Functions and procedures can be added to record types, as demonstrated in the following:

```
type
  TRec = record
    X: Integer;
    function TheAnswer: Integer;
  end;
```

Correct answer: A.

You chose: A.

☒ A. True

☐ B. False

Question 18 of 60

Procedures and Functions

Functions and procedures can be added to record types, as demonstrated in the following:

```
type  
  TRec = record  
    X: Integer;  
    function TheAnswer: Integer;  
  end;
```

Correct answer: A.

You chose: A.

✓ A. True

B. False

Exercícios (28 ao 30)