

3 - Tradução no Contexto da programação consiste na conversão de uma linguagem de alto nível para baix. Esta pode ser feita Compilador e Interpretador

4 - Sintaxe: é o conjunto de regras que definem uma linguagem de programação

- 7 - a) Variável: porção de memória usada para armazenar inf.
b) Constante: valores que não alteram durante o execução do programa
c) Identificador: é o nome pelo qual a variável é conhecida

8 - Literais; Variável

- 9 a) JAR
b) JVM
c) JDK

P. 200

10 Bytecode é código binário universal

11 JDK tools: permite a criação de programas.

12 - 1º Baixa JDK no arquiv. disponibilizado a versão Java 8; configure os variáveis PATH e Java Home a ~~versão~~

13 - As variáveis representam um espaço na memória

Paralelismo

1- Quais são as principais habilidades que deve ter um programador?
R. Auto-ensino; raciocínio lógico; capacidade de analisar, pesquisar e implementar.

2- Qual é a diferença das linguagens de programação. Faça a distinção entre LP de alto nível e de baixo nível, nas vantagens e desvantagens e exemplos de linguagens.

6- A principal diferença de linguagens de programação é de sua interpretação. A diferença de alto nível de comunicação entre computadores e humanos; diminuição tempo de Δ de um programa.

- A diferença entre LP de alto nível e baixo nível é que a linguagem de alto nível tem sintaxe voltada para o entendimento do ser humano enquanto a de máquina está voltada para o entendimento da máquina.

Alto nível: Pascal, J; J; P;

- Vantagens: fácil de aprender; produtividade; manutenção do programa é fácil;

- Desvantagens: exige mais tempo de processamento; ocupa mais memória.

Baixo nível:

- Vantagens: tempo de processamento mais rápido;

- Desvantagens

- Depende do processador

9 - Desenhe um algoritmo que determine a área de um triângulo a partir da base e altura.
Algoritmo "área-triângulo"

Var

Constante

$b = 2$; $h = 7$

Início

$A \leftarrow b \cdot h$

Exiba ("A área é igual a:", A)

Fim algoritmo

10 - Desenhe um programa que determine a área da circunferência dada $r = 2,5$.

Algoritmo "área-circunferência"

Var

A : Real

Constante

$r = 2,5$

Início

$A := \pi \cdot r^2$

Exiba ("O valor da área é:", A)

Fim algoritmo

8- Escreva um programa que permita imprimir o sucessor e antecessor de 13

Algoritmo "Antecessor e sucessor"

Var

num, ant, suc : inteiro

Inicio

num := 13

ant := num - 1

suc := num + 1

Escreva ("O antecessor do nr 13 é ", ant, " e o sucessor é ", suc)

~~leia (ant)~~

~~leia (suc)~~

Fim algoritmo

9- Uma determinada loja tem uma promoção: um desconto de 15%.
 Será aplicado a todos os produtos que custem mais de 250,00, após
 8) será aplicado a todos os outros
~~Algoritmo~~ Algoritmo «desconto loja»
 Var
 desconto, preço: real

Inicio
 escreva ("Qual é o preço do produto comprado?")
 leia (preço)
 se preço > 250,00 então
 desconto := preço - (preço * 0,15)
 escreva ("Então preço a pagar é: ", desconto) sem
 desconto ← preço - (preço * 0,08)
 escreva ("O preço a pagar é: ", desconto)
 fimse
 fim algoritmo

7^o Implemente um algoritmo que permita introduzir números no intervalo de 1 a 10 e mostre a sua representação equivalente em numeração Romana

Var

numero: inteiro

Inicio

Escreva ("Digite um n.º no intervalo de 1 à 10: ")

Leia (numero)

Escolha numero

Caso 1

Escreva ("I")

Caso 2

Escreva ("II")

Caso 3

Escreva ("III")

Caso 4

Escreva ("IV")

Caso 5

Escreva ("V")

Caso 6

Escreva ("VI")

Caso 7

Escreva ("VII")

Caso 8

Escreva ("VIII")

Caso 9

Escreva ("IX")

Caso 10

Escreva ("X")

Centina

Escreva ("Digite um n.º entre 100 e 1000: ")

Intervalo

Um algoritmo

6- Fazer o exercício anterior, considerando a soma dos pares

Algoritmo

se soma é ímpar

então

soma := 0

Exercício ("Os números pares são")

para i de 20 até 2000 passo

se $(i \bmod 2) = 0$ então

soma := soma + i

Exercício(i, ", ")

fim

fim para

Exercício("O resultado da soma é", soma)

fim Algoritmo

5. Represente um algoritmo que liste todas as raízes ímpares entre 20 e 2000 (ambos incluídos)

Var

i: inteiro

Início

Exibir ("As raízes ímpares no intervalo de 20 a 2000 são: ")

para i de 20 até 2000 passo

se $(i \bmod 2) = 0$ então

~~Exibir (i, ", ")~~

~~se~~

~~Exibir (i)~~

~~se~~

se $(i < 2000)$ então

Exibir (i, ", ")

senão

Exibir (i)

fim

fim

fim

fim algoritmo

4. Represente um algoritmo que solicite as idades de estudantes no mesmo
programação, e mostre quantos estudantes são maiores de idade.

Var

x, idade, contmaior, contmenor, conttotal: inteiro

Início

Contmaior := 0

Contmenor := 0

para x de 1 até 10 faça

escreva ("Digite idade: ")

leia (idade)

se idade \geq 18 então

Contmaior := Contmaior + 1

fimse

se idade < 18 então

Contmenor := Contmenor + 1

fimse

Exatual ("O nº de alunos maiores de idade são: "; Contmaior)

Exatual ("O nº de estudantes menores de idade são: "; Contmenor)

fimpara

Fim algoritmo

Exatual
dig nome
dos estudantes
e tem > 18 anos

Exatual
dig nome
dos estudantes
e tem > 18 anos

3. Implemente um algoritmo que permita solicitar a idade de dois amigos e posteriormente mostre como mensagem, indicando a mais velha entre eles e quantos anos de diferença há entre eles.

Var

idade1, idade2, d: inteiro

nome1, nome2: caractere

Início

Escreva ("Digite seu nome: ")

Linha (nome1)

Escreva ("Digite sua idade: ")

Linha (idade1)

Escreva ("Digite seu nome: ")

Linha (nome2)

Escreva ("Digite sua idade: ")

Linha (idade2)

se (idade1 > idade2) então

d := idade1 - idade2

Escreva ("A mais velha é", nome1, "a diferença de idade é", d)

senão

d := idade2 - idade1

Escreva ("A mais velha é", nome2, "a diferença de idade é", d)

fimse

Fim algoritmo

Valor
valor de venda, bonificação, taxa
apico, valor, inteiro

Intervalo
Intervalo ("Valor de Venda")
Intervalo ("1: 0 à 1000")
Intervalo ("2: 1000 à 5000")
Intervalo ("3: 5000 à 20000")
Intervalo ("4: 20000")

Intervalo ("Intervalo a apico com valor de Venda Correspondente:")
Linha (apico)
Intervalo ("Valor de Venda:")
Linha (valor)

Se apico = 1 inteiro

bonificacao ← valor

Se não

Se apico = 2 inteiro

bonificacao ← valor - (valor * 0,04)

Se não

Se apico = 3 inteiro

bonificacao := valor - (valor * 0,04)

Se não

Se apico = 4 inteiro

bonificacao := valor - (valor * 0,09)

Se não

Se não

Se não

Se não

Se não ("A bonificacao sera: "bonificacao")

Fim do algoritmo

Parte prática

Um estudante realiza 4 exames durante o semestre, que têm a
para 50%, 25%, 25%, 5%, respectivamente. Represente o algoritmo
correspondente para obter o nome de um estudante a partir das
notas obtidas.

Var

nome: caract

exam1, exam2, exam3, exam4, p1, p2, p3, p4, media: real

Início

Exibir ("Digite o nome do estudante: ")

ler (nome)

Exibir ("Digite a nota do primeiro exame: ")

ler (exam1)

Exibir ("Digite a nota do segundo exame: ")

ler (exam2)

Exibir ("Digite a nota do terceiro exame: ")

ler (exam3)

Exibir ("Digite a nota do quarto exame: ")

ler (exam4)

$p1 := 50/100$

$p2 := 25/100$

$p3 := 25/100$

$p4 := 50/100$

$media := (p1 * exam1 + p2 * exam2 + p3 * exam3 + p4 * exam4) / (p1 + p2 + p3 + p4)$

Exibir ("6 estudantes", nome, "tem média", media)

Exibir ("As notas inseridas foram", exam1, exam2, exam3, exam4)

Fim algoritmo