



AVALIAÇÃO

NOTA

4,9

CURSO: Ciências da Computação

6º

PERÍODO: 6º

DISCIPLINA: Paradigmas de Programação

6 Noite

TURMA:

PROFESSOR: Victor Lira

1º P1 2ª Ch P2 PS

ALUNO (A): Tarciso Dexeamps Silva

MATRÍCULA: 201310038

OBSERVAÇÕES / ORIENTAÇÃO DE PROVA

- A prova terá início às 19h, com entrada permitida até às 19h30. (só será permitida a saída da prova a partir das 19h30).
- O material de consulta, quando autorizado é de uso individual, não pode ser emprestado / compartilhado.
- As respostas à lápis não serão submetidas à revisão de prova. (o aluno deverá manter sobre a mesa apenas material autorizado).
- Não é permitido o uso de quaisquer equipamentos. (exceto calculadora).

[Texto modelo para ser editado pelo professor.]

QUESTÕES

- 0,4 1. (1,0) Explique os princípios básicos do funcionamento dos paradigmas imperativo, funcional e lógico, descrevendo qual a relação de cada um com a arquitetura de Von Neumann.
- 3,0 2. (1,0) Em relação às variáveis, explique o conceito de r-value e l-value. Além disso, forneça um exemplo.
- x 3. (1,0) Explique o conceito de Aliasing em linguagens de programação.
- 0,5 4. (1,5) Escreva uma função que recebe como parâmetro um inteiro e imprime na tela todos os números primos entre 1 e o próprio número. Obs: Utilize a linguagem de programação de sua preferência.
- 0,5 5. (2,0) Expresse em lógica de predicados:
- Pedro sabe programar em qualquer linguagem funcional existente.
 - Nenhum artista assiste a filmes que todas as pessoas que não sejam artistas gostam de assistir.
 - Não conheço quem não odeie as brincadeiras de Joãozinho.
 - Juca adormece se e somente se a luz fica acesa.
 - Ninguém visita Hermengarda, a menos que ela esteja afônica.
- 5,5 6. (2,5) Pouco se sabe da história passada da família Pinheiro. Existem alguns registros antigos que indicam que o casal José e Maria criou dois filhos, o João e a Ana. Que a Ana teve duas filhas, a Helena e a Joana, também parece ser verdade, segundo os mesmos registros. Além disso, o Mário é filho do João, pois muito se orgulha dele. Estranho também, foi constatar que o Carlos nasceu da relação entre a Helena, muito formosa, e o Mário.



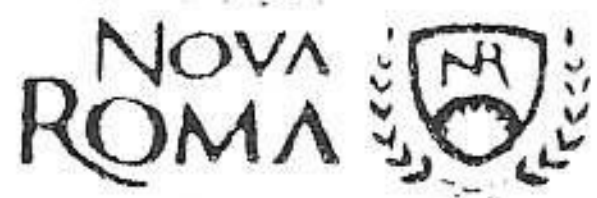
MATRÍCULA: _____

ALUNO (A): _____

- a) Utilizando o predicado progenitor(X,Y) (ou seja, X é progenitor de Y), represente em Prolog todos os progenitores da família Pinheiro.
- b) Represente em Prolog as relações: sexo (masculino ou feminino), irmã, irmão, descendente, mãe, pai, avô, tio, primo2.
- c) Formule em Prolog as seguintes questões:
1. O João é filho do José?
 2. Quem são os filhos da Maria?
 3. Quem são os primos do Mário?
 5. Quem são os ascendentes do Carlos?
 6. A Helena tem irmãos? E irmãs?

respostas na folha em anexo

Paradimas de programação - P1



NOME: Tarciso Deschamps Silva / 201310038

Data _____

1.) no imperativo ele respeita a arquitetura de Von Neumann pois o programa é carregado em memória, é enviado ao processador para alterar o estado do dado, e depois é salvo novamente em memória.

- Já no funcional ele não permite as alterações das variáveis desrespeitando a arquitetura em parte.

- e no lógico os dados carregados em memórias com suas regras e são realizadas as consultas, sem alterar os dados

→ Em todos precisamos carregar os dados na memória.

2.) l-value: indica o endereço de memória em que o dado da variável está armazenado

- r-value: representa os dados (valor) armazenados para aquela variável

Ex: l-value: 80x66

r-value: ~~80x66~~ 'A'

3.) Usar 'apelidos' para indicar um método que já existe, sem usar o mesmo nome

4.) `public int entrada()`

```
Scanner sc = new Scanner(System.in);
```

```
int valor = sc.nextInt();
```

```
return valor;
```

```
public void primos (int a) { int conta = 0;
```

```
for (int i = 1; i <= a; i++) {
```

```
if (a % i == 0) {
```

```
    conta++;
```

```
if (conta == 2) {
```

```
    System.out.println(a + " é primo");
```

```
    } else {
```

```
        System.out.println(a + " não é primo");
```

```
    }
```

```
}
```

```
public static void main (String[] args) {
```

```
    Primos p = new Primos;
```

```
    System.out.println("insira um numero");
```

```
    p.primos (p.entrada());
```

```
}
```


- 5.) a) $\exists x, \text{Pedro}(x) \rightarrow \text{sabeprogramar}(x, y) \wedge \forall y \text{ qualquer Linguagem}(y)$ X
 b) $\neg \exists x, \text{artista}(x) \wedge \text{marrista}(x, y) \wedge \text{filme}(y) \wedge \text{NãoArtistaMarrista}(y)$ X
 c) $\forall x, \text{pessoa}(x) \rightarrow \text{odeiabrincadeira}(x, y) \wedge \text{Joãozinho}(y)$ X
 d) $\exists x \text{ Juca}(x) \wedge \text{adormece}(x) \rightarrow \text{acessa}(y) \wedge \text{Luz}(y)$ X
 e) $\neg \exists x \text{ visita}(x) \rightarrow \text{afônico}(x)$ X

- 6.) a)
- | | |
|--------------------------|-----------------------------|
| progenitor (José, João) | progenitor (Ana, Helena) |
| progenitor (José, Ana) | progenitor (Ana, Joana) |
| progenitor (Maria, João) | progenitor (João, Mário) |
| progenitor (Maria, Ana) | progenitor (Helena, Carlos) |
| | progenitor (Mário, Carlos) |

- b)
- | | |
|-----------------------|-----------------------|
| masculino (José) | feminino (Ana) |
| masculino (João) | feminino (Mário) |
| masculino (Mário) | feminino (Helena) |
| masculino (Carlos) | feminino (Joana) |
| irmão (Ana, João) | irmão (João, Ana) |
| irmão (Helena, Joana) | mãe (Mário, João) |
| irmão (Joana, Helena) | mãe (Maria, Ana) |
| mãe (Ana, Helena) | mãe (Ana, Joana) |
| mãe (Helena, Carlos) | pai (José, João) |
| pai (José, Ana) | pai (João, Mário) |
| pai (Mário, Carlos) | tio (Ana, Mário) |
| tio (João, Helena) | tio (João, Joana) |
| tio (Helena) | tio (Joana, Carlos) |
| avô (José, Mário) | avô (José, Helena) |
| avô (José, Joana) | primo (Helena, Mário) |
| primo (Joana, Mário) | primo (Mário, Helena) |
| primo (Mário, Joana) | |

- c)
- 1) ? - pai (José, João) ✓
 - 2) ? - mãe (Maria, X) ✓
 - 3) ? - primo (Mário, X) ✓
 - 5) ? - pai (X, Carlos), mãe (X, Carlos) ✓
 - 6) ? - irmão (Helena, X), irmão (Helena, X) ✓