



Universidade do Vale do Itajaí - UNIVALI
Centro de Ciências Tecnológicas da Terra e do Mar
Ciência da Computação

Disciplina: Introdução à Ciência da Computação
Alunos: Gabriel Conti; Mauricio;

- 1- Linguagem de programação é o meio padronizado de gerar passos e comandos de acordo com sua semântica e sintaxe, para linguagem de máquina.
- 2- Baixo nível: interpretadas diretamente pelo computador;
Alto nível: são interpretadas por um compilador;
Funcional: é baseada em funções;
Orientada a objetos: Possui comandos específicos para classes e objetos;
Procedural: Possui funções específicas ao invés de métodos;
- 3- Linguagem de baixo nível é a linguagem mais próxima da máquina, normalmente usado para robótica ou automação, com comandos mais diretos.
- 4- É um programa que traduz os comandos escritos na linguagem Assembly para uma linguagem de máquina, a maioria dos compiladores de códigos de linguagens de alto nível fazem o processo de tradução primeiro para o Assembly e depois para a linguagem de máquina.
- 5- As linguagens estruturadas tendem a fazer com que os usuários tenham cada vez mais facilidades em comandar o computador mas elas geram códigos de baixo nível para que o hardware entenda e execute o comando desejado.
- 6- São linguagens que utilizam de rotinas ou funções para especificar os passos de um programa para que seguir ou alcançar um estado desejado, a programação procedural geralmente oferece a habilidade de reutilizar o mesmo código em diferentes lugares no programa sem copia-lo e assim diminuir o tamanho do código.
- 7- A linguagem procedural possui procedimentos e funções específicos, ao invés de “métodos” presentes na orientada a objeto;
Possui variáveis, ao invés das “instâncias de variáveis” presentes na orientada a objeto;
Contem tipos de dados definidos pelos usuários, contra as “classes”, definidas na orientada a objeto;
- 8- A programação funcional baseia-se no conceito matemático de função, em que para cada elemento do seu conjunto domínio (entrada) há apenas um elemento no seu conjunto contradomínio (saída). Normalmente são expressas por meio de outras

funções, de modo que obter o valor da função para um determinado conjunto de parâmetros envolve não só aplicar as regras daquela função, mas também fazer uso de outras funções.

- 9- Possui comandos disponíveis pela linguagem, e normalmente há comandos distintos para tratar de tipos de dados distintos;
O conceito de "objeto" é uma extensão desse comportamento para além dos tipos de dados da própria linguagem, aplicando-os também aos tipos definidos pelo usuário: Para cada estrutura de dados definida, define-se também uma série de operações que podem ser aplicadas a essa estrutura - em geral influenciadas pelo seu estado interno - e a partir de então se permite usar essas estruturas como tipos distintos, e não como estruturas de dados genéricas;
A principal diferença da orientada a objeto é a questão do estado: uma função por definição não possui efeitos colaterais, ela apenas retorna um resultado a partir de um determinado conjunto de parâmetros. Consequentemente, duas ou mais chamadas da mesma função passando os mesmos parâmetros devem produzir resultados iguais - enquanto na POO o resultado pode variar conforme o estado do objeto no momento em que a mensagem foi enviada.

10- PHP -> Notepad++

```
<html>
<head>
  <title>PHP Teste</title>
</head>
<body>
  <?php echo "<p>Olá Mundo</p>"; ?>
</body>
</html>
```

C++ - CLion

```
#include <iostream>

#include <conio2.h>

using namespace std;

int main()

{

    int x,y,z;

    x = 10;

    y = 15;

    z = x + y;
```

```
        cout << z << endl;

        return 0;

    }
```

C - Codeblocks

```
int main() {
    float nota1, nota2, media;
    nota1 = 5.5;
    nota2 = 8.5;
    media = (nota1 + nota2)/2;
    printf("Media = %f", media);
    if(media >= 7.0)
        printf(" - Aprovado");
    else // se a média é menor ou igual a 7.0
        printf(" - Prova final");

    printf("\n\n"); // salta 2 linhas em branco
    system("pause"); // não permite que a janela do prompt feche
    return 0;
}
```

Bibliografia:

<https://php.eduardokraus.com/qual-a-melhor-ide>

<http://pt.stackoverflow.com/questions/13372/programa%C3%A7%C3%A3o-funcional-e-programa%C3%A7%C3%A3o-orientada-a-objetos-o-que-s%C3%A3o-e-quais-suas>

<http://www.devmedia.com.br/programacao-orientada-a-objetos-versus-programacao-estruturada/32813>

<http://www.trabalhosfeitos.com/topicos/o-que-%C3%A9-uma-estrutura-procedural-o-que-voc%C3%AA-entende-por/0>

http://www.macoratti.net/14/06/oop_proc.htm

<http://www.criarweb.com/artigos/685.php>