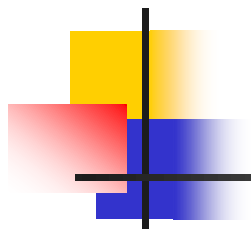




Linguagem e Paradigma de Programação

Rachel Reis
rachel@inf.ufpr.br



O que é uma Linguagem de
Programação?



Linguagem de Programação

- Instrumento usado para facilitar a comunicação entre humanos e computadores a fim de solucionar problemas.
- É composta por um conjunto de regras sintáticas e semânticas usadas para definir programas (*software*).
 - Sintaxe: representação simbólica
 - Semântica: o conceito que ela representa



Linguagem de Programação

- Exemplo:

se $x > 0$, x recebe +1

- Sintaxe: if (<expr>) <instrução>
- Semântica: se o valor da expressão for verdadeiro, a variável x é incrementada em 1.

```
if  (x > 0)
    x = x + 1;
```



Tipos de Linguagem

- 1) Linguagem de Máquina ou Baixo Nível
- 2) Linguagem de Montagem ou *Assembly*
- 3) Linguagem de Alto Nível

Tipos de Linguagem

- 1) Linguagem de Máquina ou Baixo Nível:
 - Formada por uma sequência de dígitos binários (0s e 1s)
 - Notação facilmente entendida pelo processador, mas de **difícil compreensão** para humanos.
 - Exemplo:

```
MZÀ$PÿvèšÿjË3ÀP, F  
ëfF, < uè2ÀëäÀtB*  
ÀuC+à2Àüä-I, "t" < \u€< "u-IöÄééiYË.Žt% "CÔÔÔô<i+ërâ&%. -@~Ä%v, ðvÿv  
ÿvèÄfÄÿvpÿvüèüêYY<V<FèRÿvpÿvüèWífÄ&]ËU<i fiHvW<~<F%FpÀu  
'í! '3Àé•Š~<øŠ+Ïn
```

Programação **inviável** para
seres humanos





Tipos de Linguagem

- 2) Linguagem de Montagem ou *Assembly*:
 - Formada por instruções pré-definidas que são traduções de linguagens de Alto Nível.

Tipos de Linguagem

- Exemplo:

Endereço	OPcode	Operandos
7C90EBAF	pushfd	
7C90EBB0	sub	esp, 2D0h
7C90EBB6	mov	dword ptr [ebp+FFFFFFDDCh], eax
7C90EBBC	mov	dword ptr [ebp+FFFFFFDD8h], ecx
7C90EBC2	mov	eax, dword ptr [ebp+8]
7C90EBC5	mov	ecx, dword ptr [ebp+4]
7C90EBC8	mov	dword ptr [eax+0Ch], ecx
7C90EBCB	lea	eax, [ebp+FFFFFFD2Ch]
7C90EBD1	mov	dword ptr [eax+000000B8h], ecx

Programação **viável** mas
pouco intuitiva





Tipos de Linguagem

3) Linguagens de Alto Nível:

- Aproxima-se das linguagens utilizadas por humanos para expressar problemas e algoritmos.
- Exemplo:



```
public class Retangulo
{
    private int base, altura;
    public int calculaArea()
    {
        return (base * altura);
    }
}
```



Exemplos de Linguagens

ABC, Ada, Alan, ALF, Algol, Alloy, Amiga E, AMPL, APL, AWK, B, BASIC, BCPL, BETA, Bliss, Blue, Business Rules, C, C++, Charity, CLAIRE, Clean, COBOL, COMAL, cT, DCL, Dialect, Dylan, E, Eiffel, elastiC, Elf, Erlang, Escher, Euphoria, Forth, Fortran, FPL, GNU E, Guile, Gödel, Haskell, Hugo, ICL, Icon, Inform, J, Java, Joy, Juice, K, Lava, LIFE, Limbo, LISP, LOGO, Lua, Matlab, MCPL, Mercury, Miranda, ML, Modula-2, Modula-3, NeoBook, NESL, NetRexx, Oberon, Object Oriented Turing, Objective-C, Obliq, Occam, Octave, Oz, Pascal, Perl, Phantom (Phi), PHP, Pike (LPC), PiXCL, PL/B, PL/I, Pliant, Postscript, Prolog, Python, R, REBOL, Rexx, RPG, RPL/2, Ruby, S, Sather, Scheme, Self, SETL, Simula, Sisal, Smalltalk, SNOBOL, SR, TADS, Tcl, Theta, TOM, V, Visual Basic, Yorick, ZPL



Por que existem várias Linguagem de
Programação?



Domínios de Programação

- Os programas de computadores são aplicados a uma infinidade de áreas diferentes.
- Linguagens de Programação com objetivos diferentes têm sido desenvolvidas
- Áreas de Aplicação:
 - Aplicações científicas
 - Aplicações comerciais
 - Inteligência artificial
 - Desenvolvimento de *software* básico
 - Desenvolvimento de *software* para web
 - etc.

Algumas Linguagens Malucas

- Brainfuck (Hello World):

```
+++++++ [ >++++ [ >+>++++>++++>+<<<<- ] >+>+>-  
>>+ [<] <- ] >> .>--- .+++++++ . .+++ .>> .<-  
.< .+++ .----- .----- .>>+ .>+ .
```

- ArnoldC (Hello World):



```
IT'S SHOWTIME  
TALK TO THE HAND "Hello World!"  
YOU HAVE BEEN TERMINATED
```

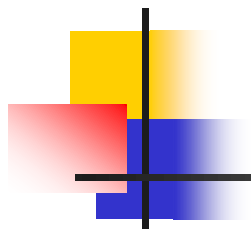




Algumas Linguagens Malucas

- Ook! (Hello World):

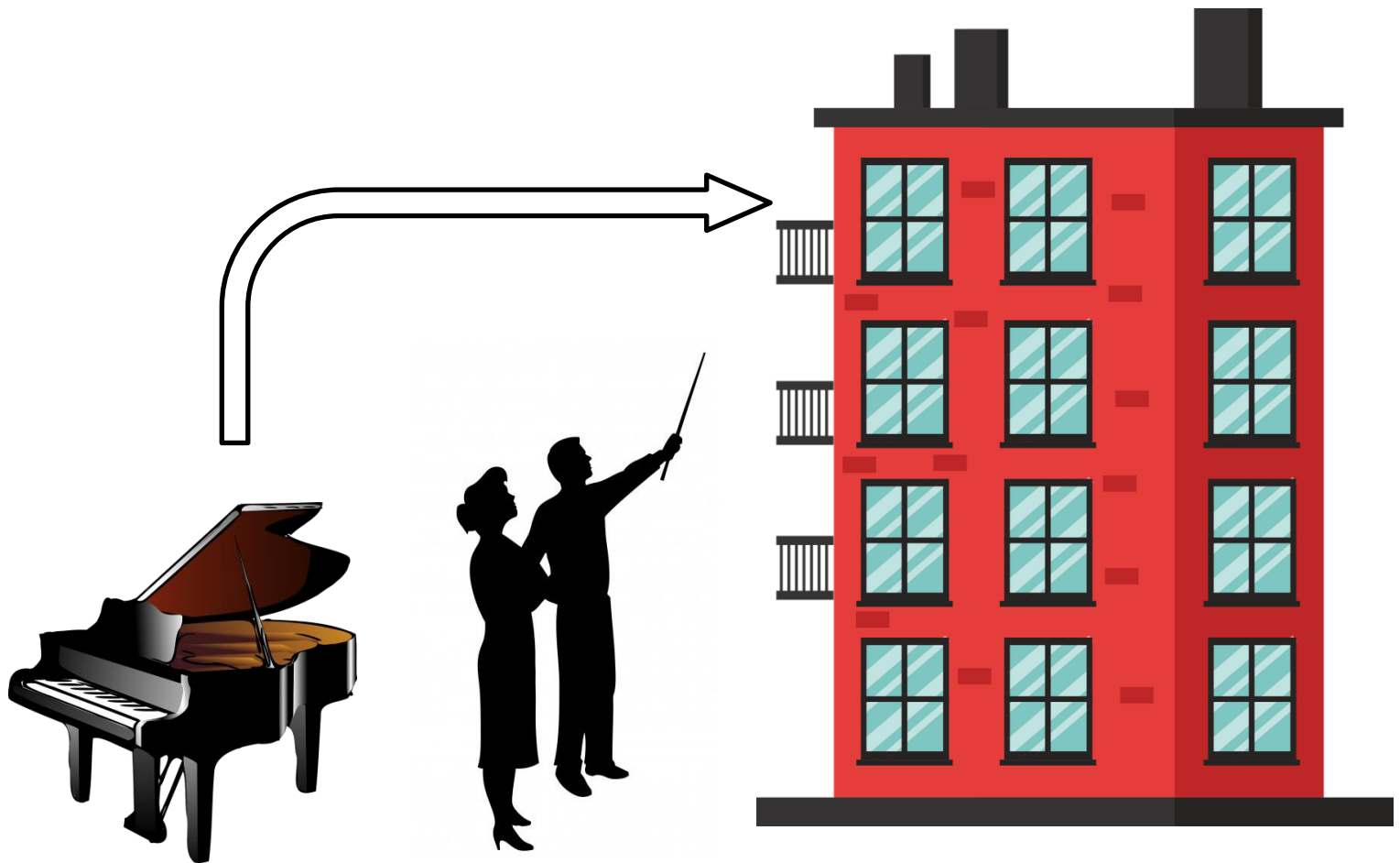
```
Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook? Ook! Ook! Ook? Ook! Ook? Ook.
Ook! Ook. Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook?
Ook! Ook! Ook? Ook! Ook? Ook. Ook. Ook. Ook! Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook. Ook! Ook. Ook. Ook. Ook.
Ook. Ook. Ook! Ook. Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook? Ook! Ook! Ook? Ook! Ook? Ook. Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook.
Ook? Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook! Ook. Ook. Ook. Ook. Ook. Ook.
Ook! Ook. Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook.
Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook!
Ook! Ook. Ook. Ook? Ook. Ook? Ook. Ook. Ook. Ook! Ook.
```



O que são Paradigmas de
Programação?

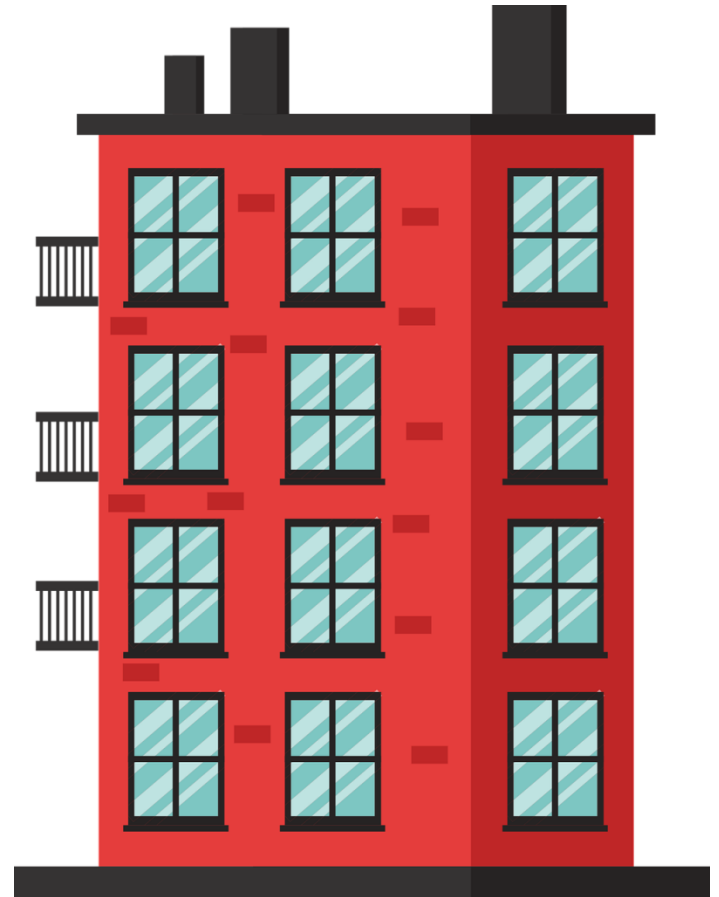


Situação-problema



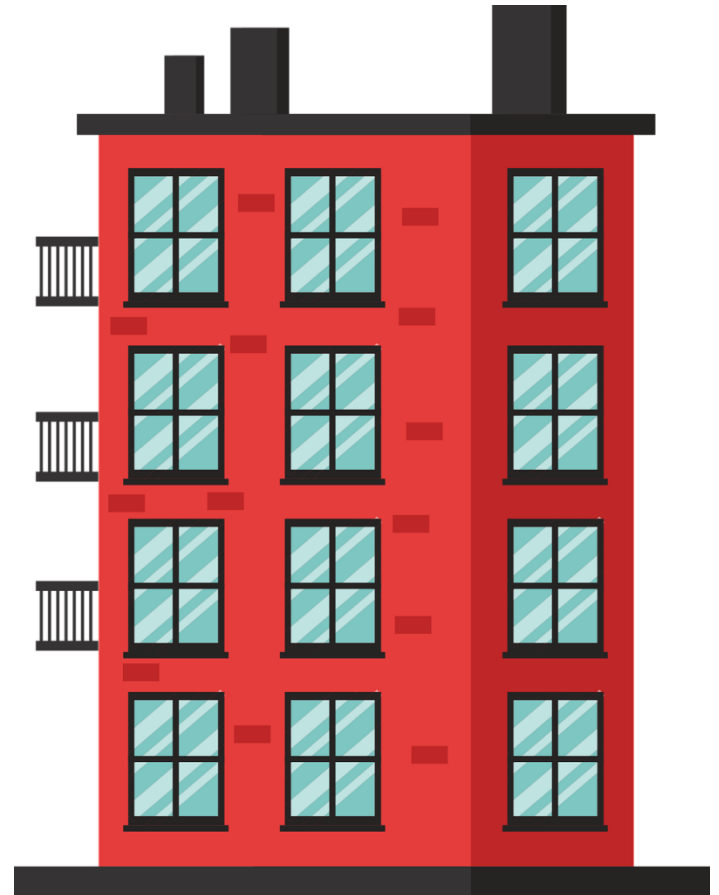


Paradigma 1





Paradigma 2





Paradigma 3

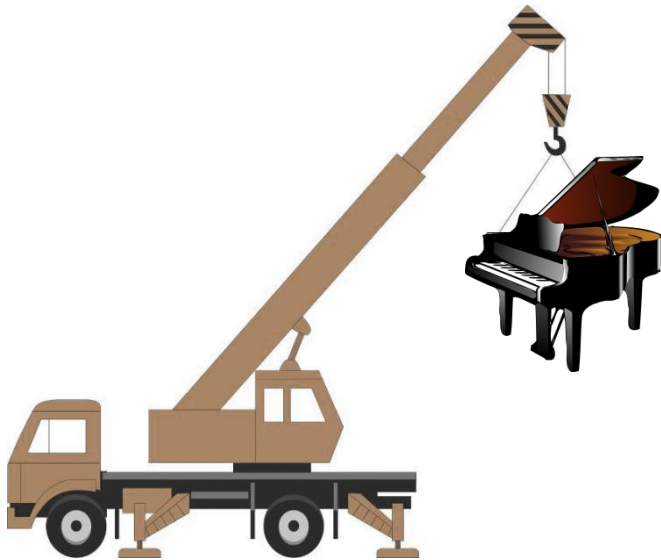


Paradigma 4





Melhores Paradigmas



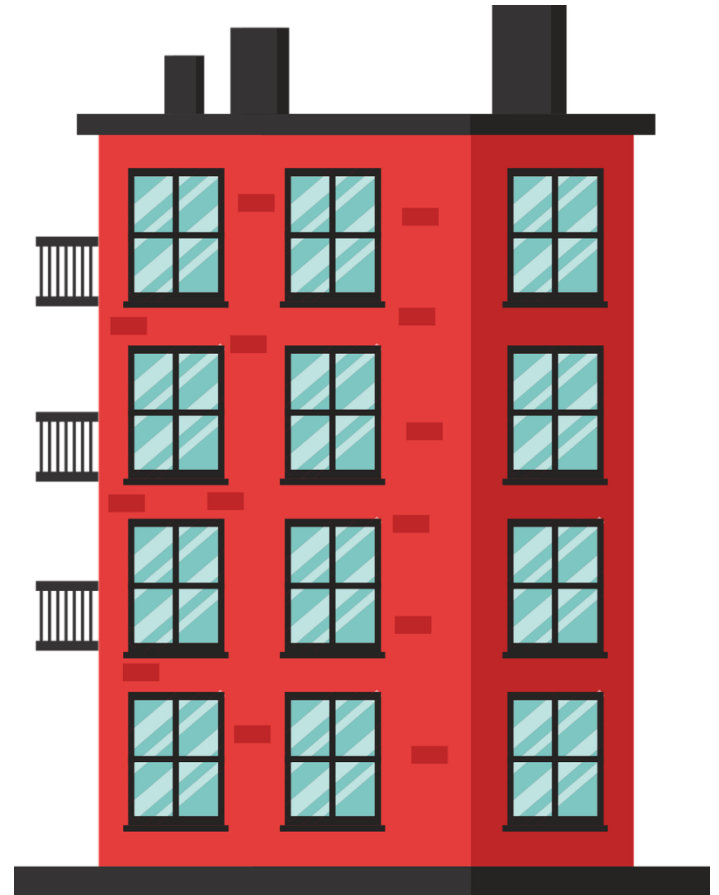
Paradigma 3



Paradigma 4



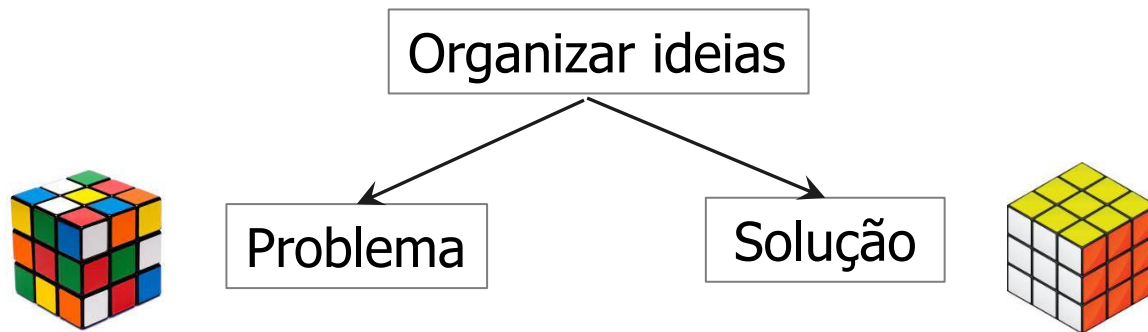
Paradigma 5





Paradigmas de Programação

- Estilo de programação, a forma como o programador descreve a solução computacional de um problema.





Paradigma de Programação

- Cada linguagem apresenta uma maneira particular de modelar o que é um programa.
- A escolha de um determinado paradigma influencia a forma com que uma aplicação real é modelada do ponto de vista computacional.



Paradigmas de Programação

Paradigmas:

- imperativo
- estruturado



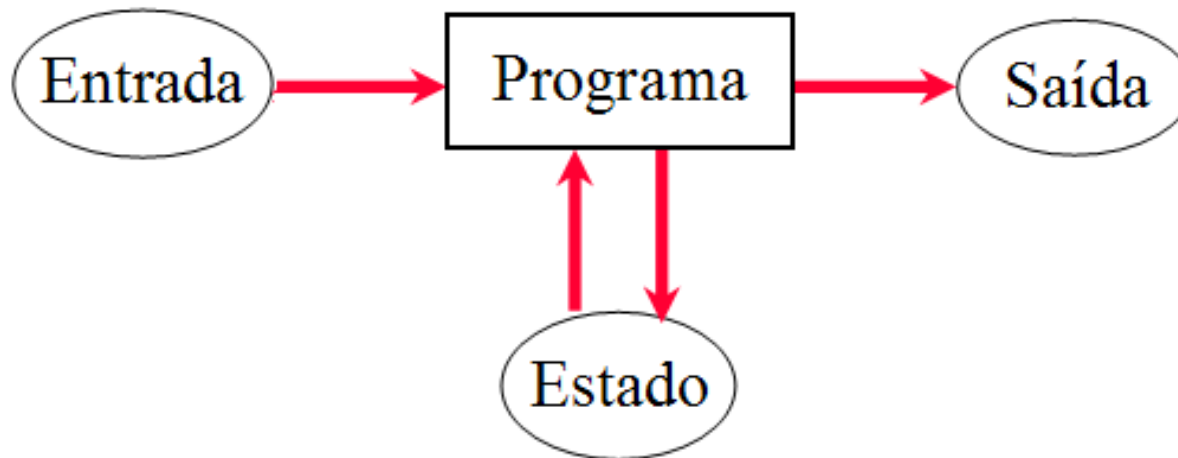


Paradigma Imperativo

- Instruções passadas ao computador devem especificar o passo a passo das ações que serão executadas.
- Essas ações mudam o estado do programa a partir de variáveis que ficam armazenadas na memória até atingir um estado final.
- Linguagens que tem como base o paradigma imperativo: Fortran, Pascal, Cobol, C



Paradigma Imperativo





Paradigma Imperativo

```
int main()
{
    int num1, num2, soma;
    printf("Digite dois números: ");
    scanf("%d, %d" , &num1, &num2);
    soma = num1 + num2;
    printf("Soma: ", soma);

    return 0;
}
```



Paradigma Estruturado

- As instruções passadas ao computador podem ser formadas por três estruturas:
 - Sequência: instruções são codificadas em determinada ordem.
 - Estruturas condicionais: if/else, switch, etc.
 - Estruturas de repetição: for, while, do-while, etc.
- Também conhecido como procedural, capacidade de modularização (métodos, funções, rotinas).



Paradigma Estruturado

```
int soma (int n1, int n2){
    return n1 + n2;
}
int main(){
    int num1, num2, resp;
    do{
        printf("Digite dois números: ");
        scanf("%d, %d" , &num1, &num2);
    }while(num1 < 0 || num2 < 0)
    resp = soma(num1, num2);
    printf("Soma: ", resp);
    return 0;
}
```

Paradigmas de Programação

Paradigmas:

- **imperativo**
- **estruturado**

Por que aprender
esses paradigmas?



Computadores seguem
sequências de instruções



Paradigmas de Programação

- Sobre os diferentes paradigmas...

Paradigmas tradicionais:

- imperativo
- estruturado
- **orientado a objetos**



Paradigma Orientado a Objetos

- Ao invés de programar pensando como máquina, pode-se programar pensando como humanos.
- Organização do código por meio de objetos que contêm dados, estados próprios e métodos que alteram e recuperam os dados/estados.
- Os objetos comunicam entre si para compor a lógica do programa.
- Exemplos de linguagem
 - Java, C++, Simula, Smaltalk, C#



Paradigmas de Programação

- Sobre os diferentes paradigmas...

Paradigmas tradicionais:

- imperativo
- estruturado
- orientado a objetos

o que têm em
comum?



Descrevem para o computador:

- **O** que fazer
- **Como** fazer

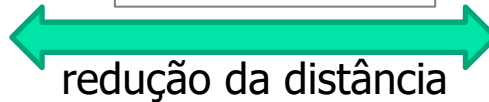


Paradigmas de Programação

Paradigmas tradicionais:

- imperativo
- estruturado
- orientado a objetos

evolução dos
compiladores



Outros paradigmas:

- **declarativo**
- **lógico**
- **funcional**



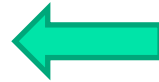
Paradigmas de Programação

- Sobre os diferentes paradigmas...

Descrevem para o computador:

- **O** que fazer

o que têm em
comum?



Outros paradigmas:

- declarativo
- lógico
- funcional



Paradigmas de Programação

- Paradigma declarativo
 - O fluxo lógico é implícito.
 - Separação clara entre o que o programador deseja obter do como proceder para alcançar o que se deseja.
 - Linguagens de alto nível que permitem aos programadores dizer apenas o que desejam.

Paradigmas de Programação

- Paradigma declarativo (exemplo SQL)

```
SELECT      UPPER(nome)
FROM        alunos
WHERE       nota >= 7
ORDER BY    nome
```





Paradigmas de Programação

- Paradigma lógico

- Especifica-se apenas fatos e regras de inferência.
- O objetivo (retorno) é escrito em forma de pergunta.

- Exemplo

```
aprovado(X) :- nota(X, N), N >= 7.  
  
sort(  
    findall(Alunos, aprovados(Alunos), Aprovados)  
)
```



Paradigmas de Programação

- Paradigma funcional

- Descreve a computação como uma expressão a ser avaliada.
- A principal forma de estruturar o programa é pela definição e aplicação de funções.

- Exemplo

```
sort[upper(nome aluno) | aluno <- alunos, nota aluno >= 7]
```



Paradigmas de Programação

- Muitas linguagens de programação são, na realidade, **multi –paradigmas**.
 - Contém características de diversos paradigmas.
 - Na prática, elas acabam favorecendo um paradigma específico recebendo, por exemplo, o título de linguagem “funcional” ou “orientada a objetos”.



Paradigma nessa Disciplina

- Imperativo
- Estruturado
- Declaratio
- Lógico
- **Funcional**
- **Orientado a Objetos**



Referência

- Conceitos de Linguagens de Programação. Sebesta, R. S.. 6a. ed.. Bookman. São Paulo, 2008.