

APRESENTAÇÃO DA DISCIPLINA

MC102 - Algoritmos e
Programação de
Computadores

Santiago Valdés Ravelo
[https://ic.unicamp.br/~santiago/
ravelo@unicamp.br](https://ic.unicamp.br/~santiago/ravelo@unicamp.br)


03/24

0



UNICAMP



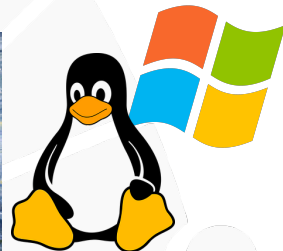


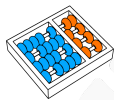
“Ciência da computação tem tanto a ver com o computador como a astronomia com o telescópio, a biologia com o microscópio, ou a química com os tubos de ensaio. A Ciência não estuda ferramentas, mas o que fazemos e o que descobrimos com elas.”

Atribuída a Edsger W. Dijkstra.



Áreas de aplicação da computação





O que faz um computador?

- ▶ Executa programas?
- ▶ O que são programas?
- ▶ Como os programas são criados?



ALGORITMOS



O que é um algoritmo?

Um **ALGORITMO** pode ser descrito, informalmente, como uma **SEQUÊNCIA LÓGICA, FINITA E DEFINIDA DE AÇÕES** capazes de resolver um problema.

O conceito de algoritmo vai além da programação.



Algoritmos em tarefas comuns

- ▶ Mascar chiclete.
- ▶ Fritar um ovo.
- ▶ Trocar pneu furado.
- ▶ Trocar uma lâmpada queimada.
- ▶ Calcular a média de duas notas.



Algoritmos em tarefas comuns. Sugestões

Mascar chiclete:

1. Pegar o chiclete.
2. Retirar o papel.
3. Jogar o papel no lixo.
4. Colocar o chiclete na boca.
5. Enquanto o chiclete tiver sabor:
Mastigar o chiclete.
6. Jogar o chiclete no lixo.



Algoritmos em tarefas comuns. Sugestões

Calcular a média de duas notas:

1. Ler a primeira nota.
2. Ler a segunda nota.
3. Somar as duas notas.
4. Dividir a soma por 2.
5. Imprimir o resultado da divisão.



O que é imprescindível na proposta de um algoritmo?

- ▶ Compreender os requisitos do problema.
- ▶ Identificar as entradas.
- ▶ Identificar as saídas.
- ▶ Definir uma sequência lógica e finita de ações para solucionar o problema.
- ▶ Garantir clareza e precisão dessas ações.
- ▶ Garantir que as saídas sejam corretas.

O que acontece quando não verificamos esses itens?



O algoritmo pode ser escrito também em pseudocódigo (mais comum) ou com um fluxograma.



Pseudocódigo

Um pseudocódigo é uma forma abstrata de escrever o programa

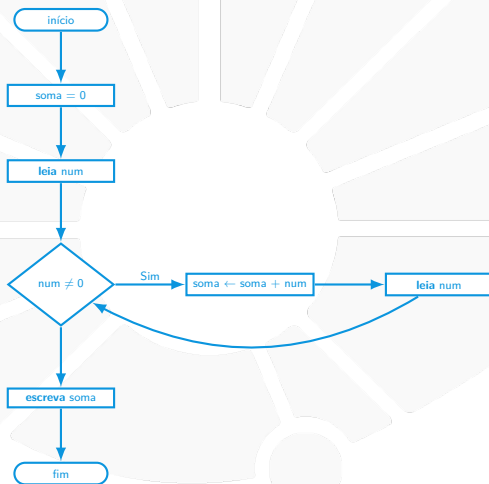
- ▶ Não estamos preocupados com a linguagem.
- ▶ Mas sim com a solução do problema.
- ▶ Também com a clareza.

```
1 soma ← 0
2 leia num
3 enquanto num ≠ 0
4   soma ← soma + num
5   leia num
6 escreva soma
```



Fluxograma

A ideia é representar as instruções graficamente:





PROGRAMAÇÃO



Programa

Um **PROGRAMA** é uma sequência de instruções que especificam como executar uma computação

- ▶ Ex: calcular a soma de números até que o zero seja digitado.
- ▶ É escrito em uma determinada linguagem:
 - ▶ Chamada de linguagem de programação.
 - ▶ Ex: Python, C, C++, Java, etc. . . .
- ▶ O texto do programa é chamado de **CÓDIGO FONTE**.



O que é uma linguagem de programação?

- ▶ Meio de comunicação:



Indivíduo que deseja
solucionar um problema



Computador

- ▶ Deve **LIGAR** o **PENSAMENTO HUMANO** com a **PRECISÃO** requerida para o **PROCESSAMENTO DA MÁQUINA**.



Execução de programas

Código-fonte



Compilador



Código-objeto

Código escrito em uma LP
(Programa)

Código de máquina
(Executável)



Como classificar LPs?

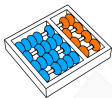
- ▶ Em relação ao **PARADIGMA**:
 - ▶ Imperativo, funcional, lógico, orientado a objetos etc.
- ▶ Em relação ao **NÍVEL**:
 - ▶ Baixo, Médio ou Alto.



O que é o paradigma?

MODELO INTERPRETATIVO ou conceitualização de uma **REALIDADE**.

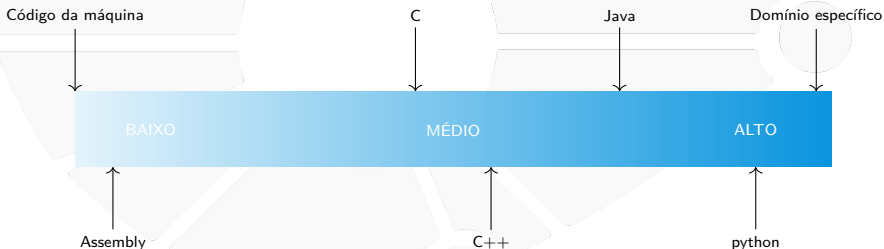
Um **PONTO DE VISTA** que determina como uma **REALIDADE** é entendida.



O que é o nível?

O **NÍVEL** de uma LP é um indicativo da **CAPACIDADE DE ABSTRAÇÃO** em relação ao hardware e o código da máquina que a linguagem oferece ao programador.

A classificação por nível pode ser considerada como um espectro:





Exemplo de duas linguagens diferentes

Python

```
1 soma = 0
2 num = int(input())
3 while(num != 0):
4     soma = soma + num
5     num = int(input())
6 print(soma)
```

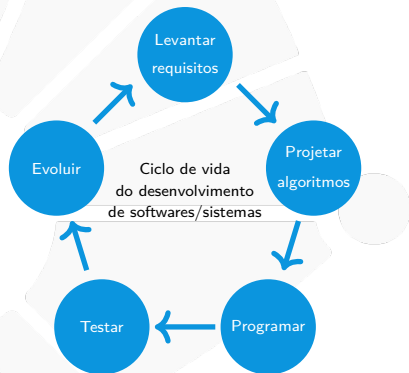
C

```
1 #include <stdlib.h>
2 int main() {
3     int soma = 0, num;
4     scanf("%d", &num);
5     while(num != 0){
6         soma = soma + num;
7         scanf("%d", &num);
8     }
9     printf("%d", soma);
10    return 0;
11 }
```



Fases para elaborar um programa

- ▶ Coleta, análise e especificação de requisitos.
- ▶ Algoritmo.
- ▶ Implementação.
- ▶ Teste.
- ▶ Manutenção.





A DISCIPLINA



Objetivos

Introduzir o uso do computador na resolução de problemas através de desenvolvimento de programas em linguagens de alto nível.

Para isto faremos:

- ▶ Análise do problema.
- ▶ Escolha de uma solução.
- ▶ Projeção de um algoritmo para a solução escolhida.
- ▶ Implementação do algoritmo na linguagem de programação **Python**.
- ▶ Teste e execução da solução implementada.



Objetivos

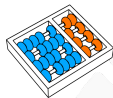
Por que Python?

- ▶ Linguagem de programação de alto nível, muito intuitiva.
- ▶ Popular e com constante crescimento na comunidade:
 - ▶ 1^{ra} no índice TIOBE [↗](#).
 - ▶ 2^{da} no GitHub [↗](#).

Estamos interessados não apenas em aprender Python:

- ▶ Mas aprender a **PROJETAR ALGORITMOS!**
- ▶ Você ainda será um programador se Python ficar obsoleto!

O algoritmo não depende da linguagem de programação! Pode ser usado e reutilizado nas mais variadas linguagens!



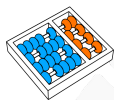
Carga horária

6 horas semanais divididas em três encontros de 2 horas cada:

- ▶ 1 encontro com aulas teóricas nas terças.
- ▶ 1 encontro com aulas teóricas nas quintas.
- ▶ 1 encontro com aulas práticas nas sextas.

IMPORTANTE:

- ▶ A aprovação depende de 75% de presença!!!
- ▶ Dedique mais tempo, 6 horas semanais não é suficiente!!
- ▶ Não se conforme com entregar o mínimo solicitado!



Avaliação. Cálculo da média final

Itens que serão avaliados:

(L) Laboratórios.

(P) Projetos.

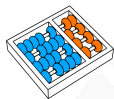
(T) Testes.

(D) Dúvidas de aula.

Cálculo da média final:

$$M = \begin{cases} \min \left(L, \frac{P+T+D}{3} \right), & \text{se } L < 5 \text{ ou } P + T + D < 10 \\ \frac{6,0L+2,5P+1,0T+0,6D}{10}, & \text{caso contrário.} \end{cases}$$

Alunos com $0 \leq M < 5$ e pelo menos 75% de frequência poderão fazer entregas atrasadas com desconto de 50% do valor.



Ferramentas online



▶ Google Sala de Aula

▶ beecrowd

▶ Página da disciplina

APRESENTAÇÃO DA DISCIPLINA

MC102 - Algoritmos e
Programação de
Computadores

Santiago Valdés Ravelo
[https://ic.unicamp.br/~santiago/
ravelo@unicamp.br](https://ic.unicamp.br/~santiago/ravelo@unicamp.br)

03/24

0



UNICAMP

