

# ALGORITMOS

Felipe Pereira de Oliveira

SITUAÇÃO PROBLEMA

# SITUAÇÃO PROBLEMA – Quem é o maior?

Apresentação da Situação Problema



Como sei qual destas pessoas é a maior?



altura1  
altura2  
altura3  
altura4  
altura5  
altura6

Neste exemplo as pessoas se encontram desorganizadas. Em programação seria como a altura de cada pessoa fosse designada por uma variável específica.  
Primeiramente organizar estas pessoas. **Mas como?**

# SITUAÇÃO PROBLEMA – Quem é o maior?

1º Passo: Coloca-las em fila



Desta forma saberemos localizar qual deles é o maior.



O equivalente em programação seria coloca-las em um vetor

alturas : vetor[0..9] de real

1,75	1,68	1,81	1,74	1,72	1,70	1,65	1,72	1,75	1,76
0	1	2	3	4	5	6	7	8	9

# SITUAÇÃO PROBLEMA – Quem é o maior?

Tudo Organizado. E agora?

Nós sabemos, tendo os valores das alturas de todos os indivíduos da fila, quem é o maior.



1,75	1,68	1,81	1,74	1,72	1,70	1,65	1,72	1,75	1,76
0	1	2	3	4	5	6	7	8	9

# SITUAÇÃO PROBLEMA – Quem é o maior?

Tudo Organizado. E agora?

Mas e um computador? Como ele saberia qual dos valores é o maior?

Para conseguir tal proeza devemos  
construir em um software um

# ALGORITMO



# SITUAÇÃO PROBLEMA – Quem é o maior?

Tudo Organizado. E agora?

Nós sabemos, tendo os valores das alturas de todos os indivíduos da fila, quem é o maior.



1,75	1,68	1,81	1,74	1,72	1,70	1,65	1,72	1,75	1,76
0	1	2	3	4	5	6	7	8	9

ALGORITMO

# ALGORITMO

## Definição

*"Um algoritmo é uma sequência de passos para solucionar um problema";*

*"Algoritmo é um conceito. Ele não resolve o problema em si. Mas quem segue seus passos (quem o implementa) consegue resolvê-lo";*

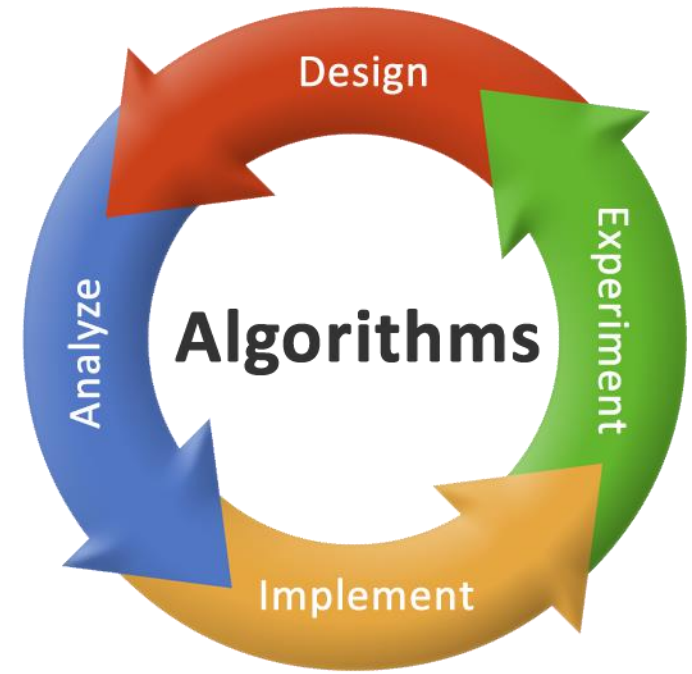
*"Algoritmos são planos a ser seguidos"*

# ALGORITMO

## Definição

Em nosso curso aprenderemos algoritmos computacionais com soluções em programação.





# CONSTRUINDO UM ALGORITMO

# CONSTRINDO UM ALGORITMO


## Motivação!

Só conseguiremos realizar a identificação da maior pessoa da fila se desenvolvermos um algoritmo para definir quais passos tomar.

Para criar nosso algoritmo vamos identificar as possíveis soluções analisando o problema.

# CONSTRUINDO UM ALGORITMO

## 1º Analisando o Problema

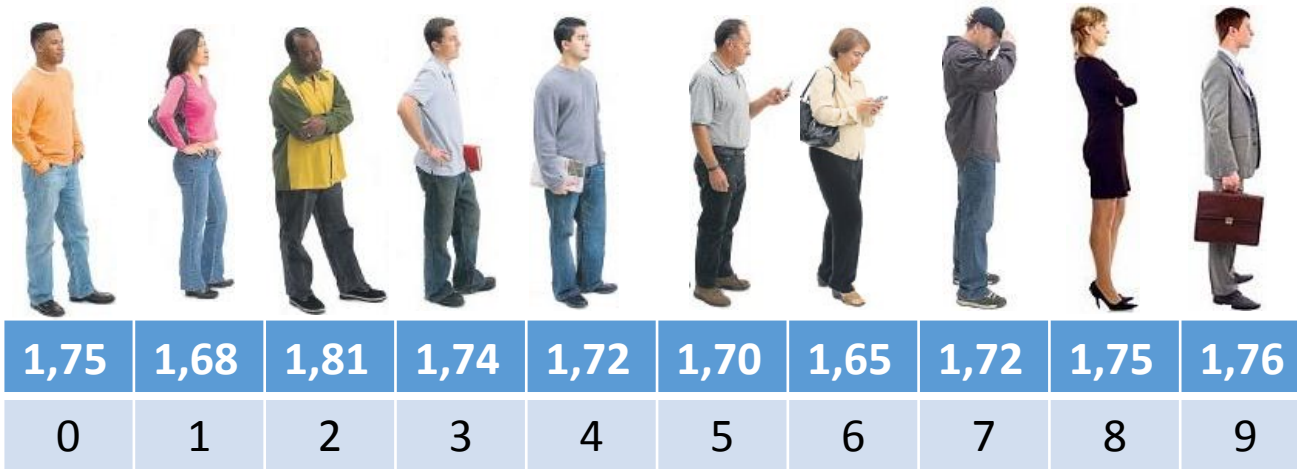


1,75	1,68	1,81	1,74	1,72	1,70	1,65	1,72	1,75	1,76
0	1	2	3	4	5	6	7	8	9

Perceba que a altura das pessoas **já está armazenada no vetor**. Mas elas estão desorganizadas. Por este motivo, teremos que comparar a altura de **todas as pessoas** para saber qual delas é a maior.

# CONSTRUINDO UM ALGORITMO

## 1º Analisando o Problema



Primeiramente, vamos começar pelo primeiro elemento do vetor. Como eu ainda não conheço os outros elementos eu sempre considerarei que



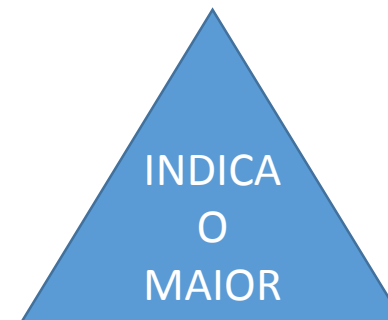
# CONSTRUINDO UM ALGORITMO

## 1º Analisando o Problema

1. O primeiro elemento será sempre considerado o maior;

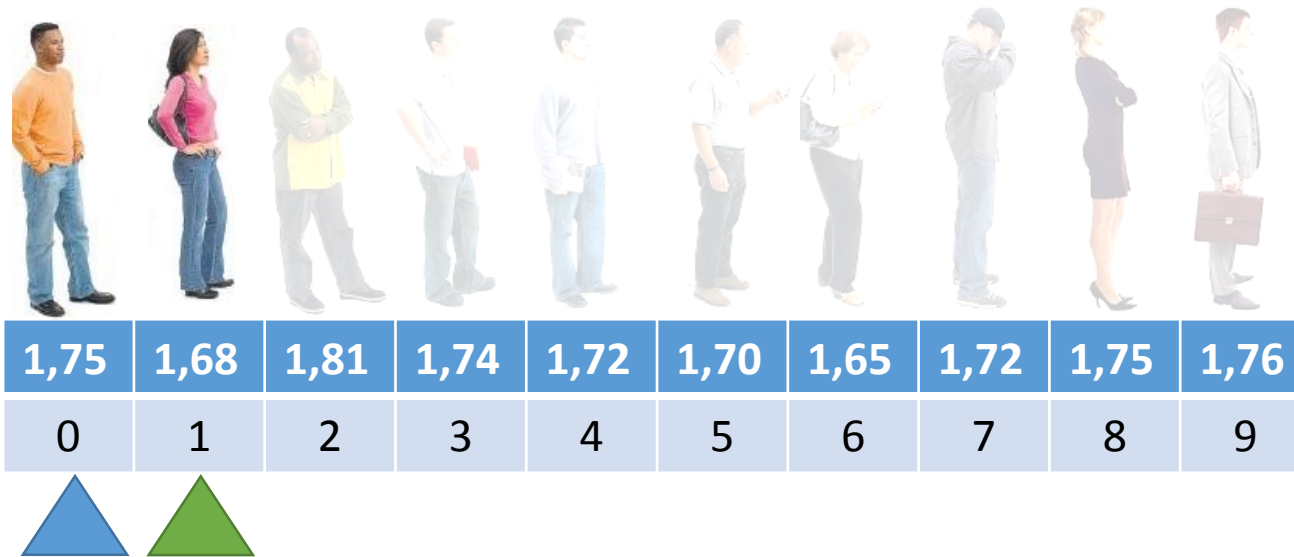


Primeiramente, vamos começar pelo primeiro elemento do vetor. Como eu ainda não conheço os outros elementos eu sempre considerarei que o primeiro elemento é a maior pessoa.



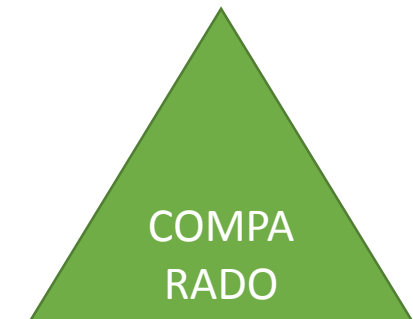
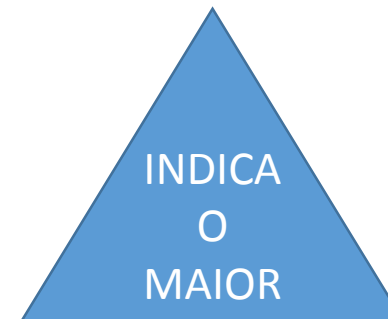
# CONSTRUINDO UM ALGORITMO

## 1º Analisando o Problema



Compara o elemento **marcado como maior** com o elemento **comparado** atual

1. O primeiro elemento será sempre considerado o maior;
2. Se houver elementos para ser comparados, vai para o passo 3, caso contrário, vai para o passo x;
3. Se o elemento **comparado** for maior que o **elemento marcado como maior**, troca pelo **comparado**;



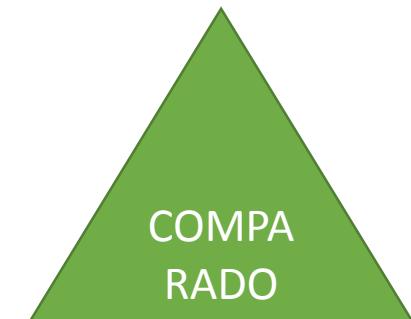
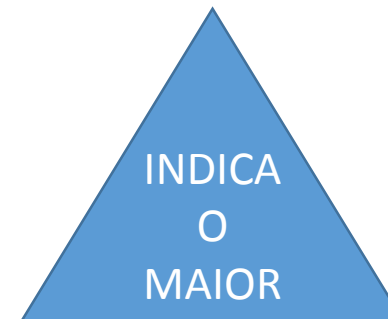
# CONSTRUINDO UM ALGORITMO

## 1º Analisando o Problema



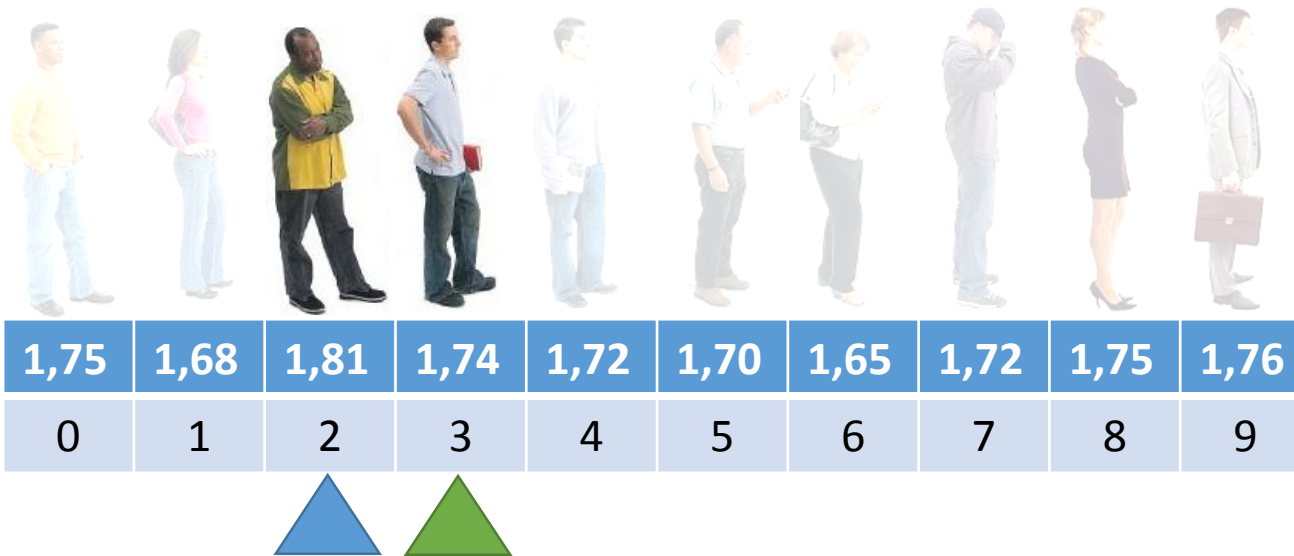
Se o **comparado** for maior que o **marcado como maior**, troca.

1. O primeiro elemento será sempre considerado o maior;
2. Se houver elementos para ser comparados, vai para o passo 3, caso contrário, vai para o passo 5;
3. Se o elemento **comparado** for maior que o **elemento marcado como maior**, troca pelo **comparado**;
4. Vai para o passo 2;
5. Exibe quem é **marcado como maior**;



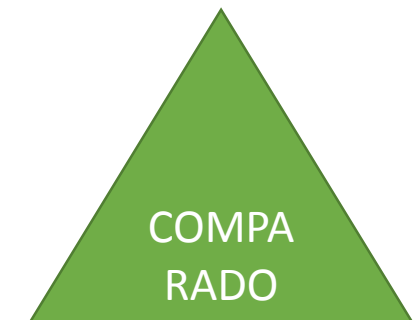
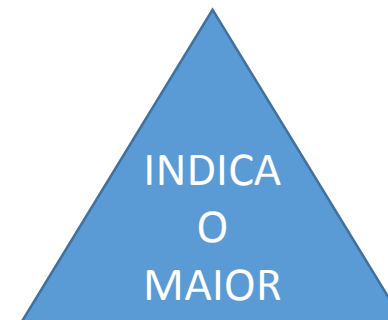
# CONSTRUINDO UM ALGORITMO

## 1º Analisando o Problema



Continua comparando para checar se não existem ninguém maior.

1. O primeiro elemento será sempre considerado o maior;
2. Se houver elementos para ser comparados, vai para o passo 3, caso contrário, vai para o passo 5;
3. Se o elemento **comparado** for maior que o **elemento marcado como maior**, troca pelo **comparado**;
4. Vai para o passo 2;
5. Exibe quem é **marcado como maior**;



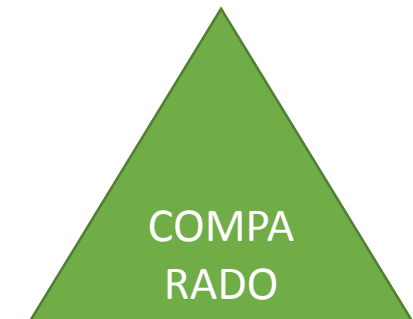
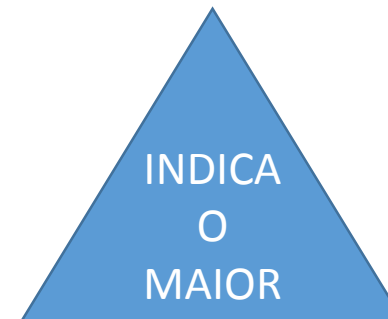
# CONSTRUINDO UM ALGORITMO

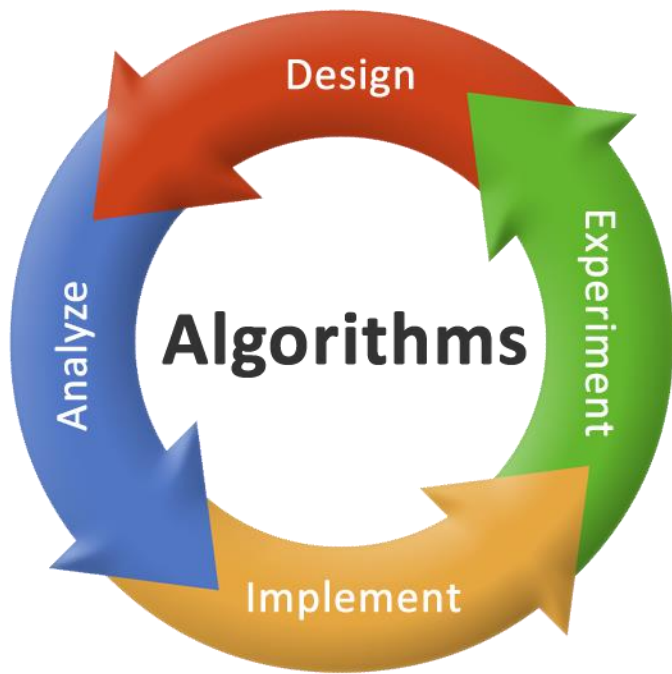
## 1º Analisando o Problema



Finalmente, exibe qual deles foi o maior.

1. O primeiro elemento será sempre considerado o maior;
2. Se houver elementos para ser comparados, vai para o passo 3, caso contrário, vai para o passo 5;
3. Se o elemento **comparado** for maior que o **elemento marcado como maior**, troca pelo **comparado**;
4. Vai para o passo 2;
5. Exibe quem é **marcado como maior**;





Implemente o algoritmo na linguagem escolhida pelo professor

1. O primeiro elemento será sempre considerado o maior;
2. Se houver elementos para ser comparados, vai para o passo 3, caso contrário, vai para o passo 5;
3. Se o elemento **comparado** for maior que o **elemento marcado como maior**, troca pelo **comparado**;
4. Vai para o passo 2;
5. Exibe quem é **marcado como maior**;