



Ciência da Computação  
Algoritmos e Estrutura de Dados 1

# Entrada, Processamento e Saída

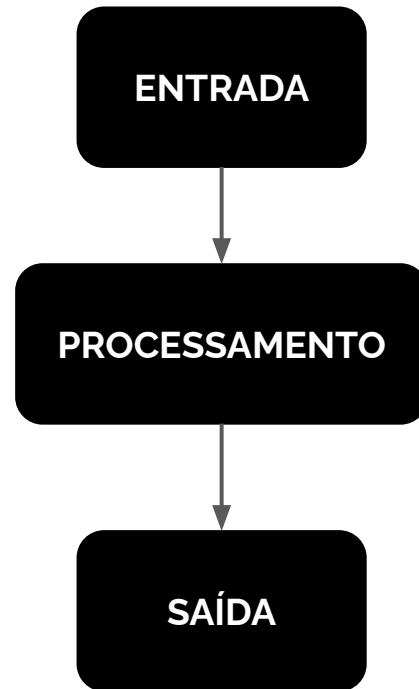
---

## Revisão

Rafael Liberato  
[liberato@utfpr.edu.br](mailto:liberato@utfpr.edu.br)

# **Modelo Entrada-Processamento-Saída**

---



# Modelo Entrada-Processamento-Saída

## Significado

### Cálculo do IMC

ENTRADA

2

**O QUE**

eu preciso /  
eu tenho

Altura e peso

2

PROCESSAMENTO

3

**COMO**

fazer

Aplicar a fórmula para efetuar o  
cálculo do índice e identificá-lo na  
tabela do IMC.

3

SAÍDA

1

**O QUE**

eu quero

Determinar uma classificação sobre  
o peso de um indivíduo com base  
no Índice de Massa Corporal (IMC)

1

# Modelo Entrada-Processamento-Saída

## Significado

ENTRADA

PROCESSAMENTO

SAÍDA

2

**O QUE**

eu preciso /  
eu tenho

3

**COMO**

fazer

1

**O QUE**

eu quero

## Encontrar um elemento em um vetor

Conjunto dos valores inteiros e o valor a ser encontrado

2

Comparar o valor a ser encontrado com cada elemento do conjunto até encontrá-lo.

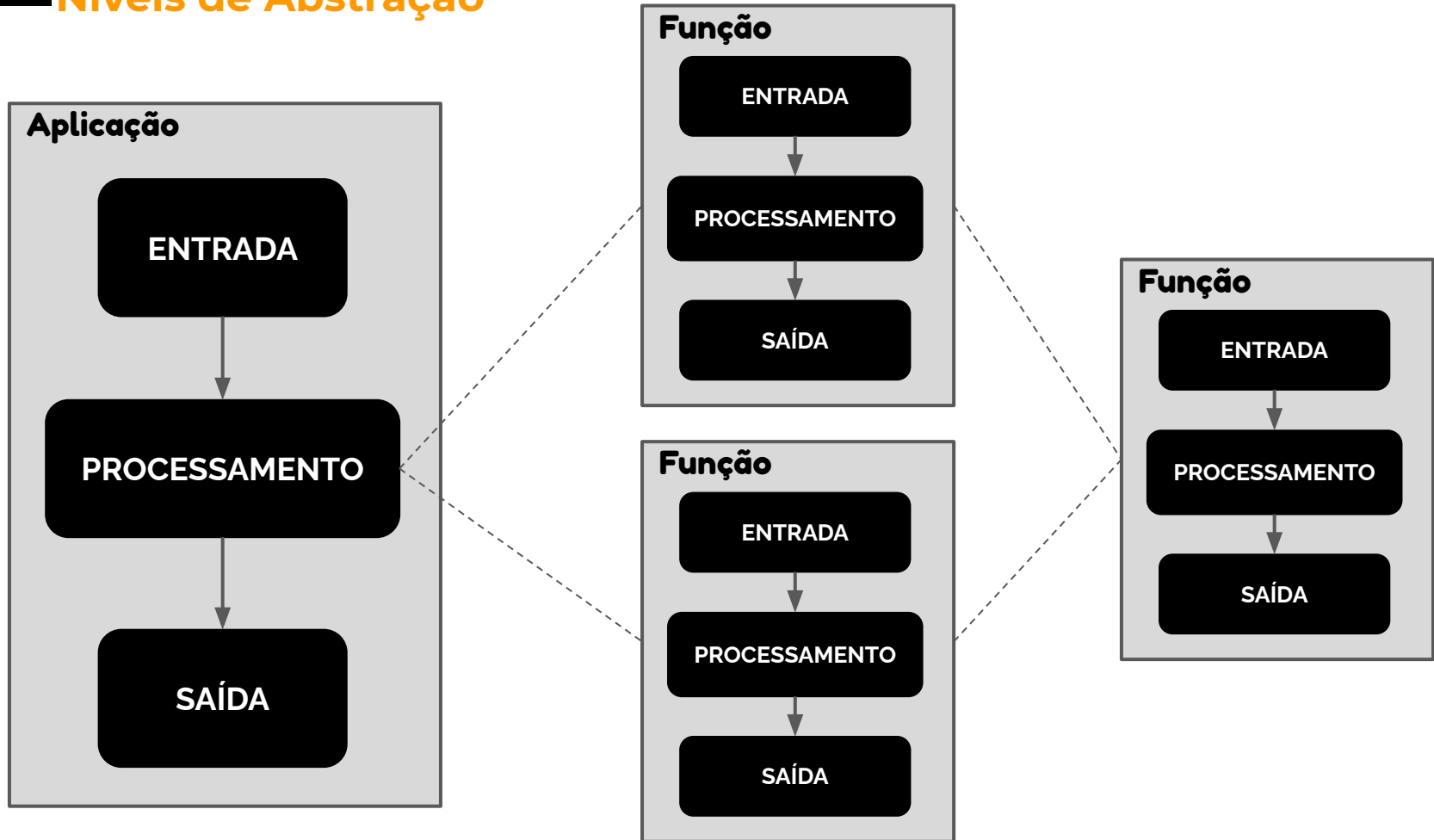
3

Verificar se um determinado valor está contido em um conjunto de números inteiros

1

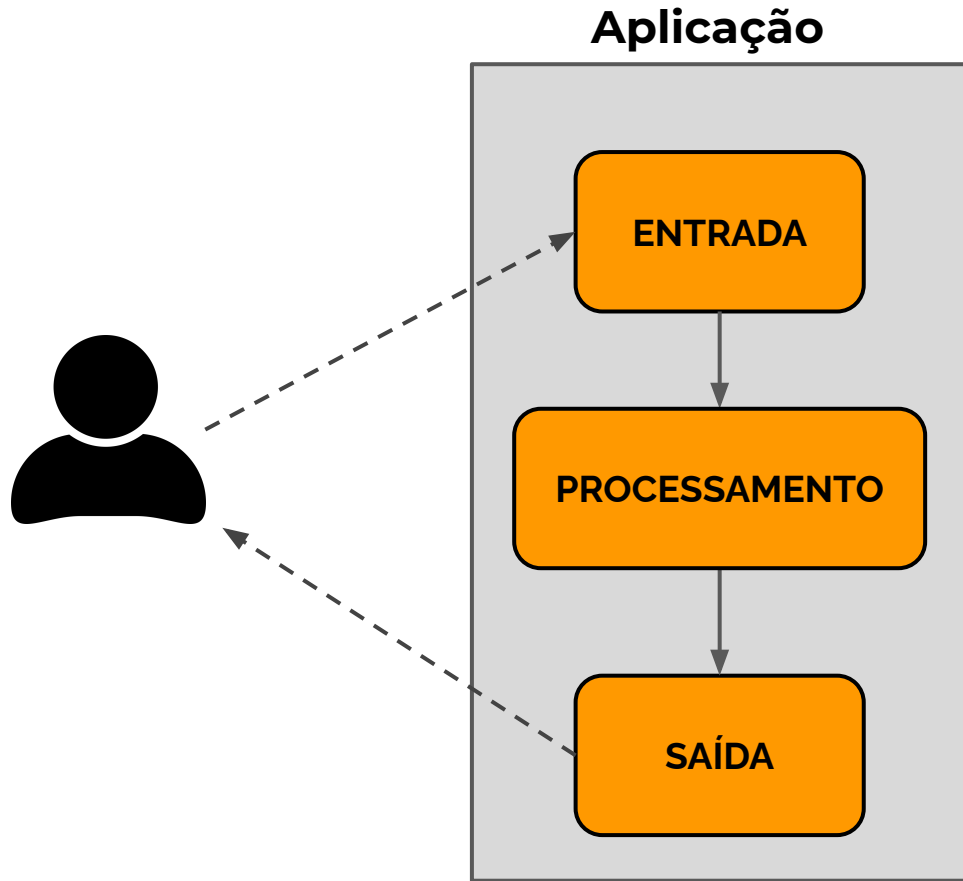
# Modelo Entrada-Processamento-Saída

## Níveis de Abstração



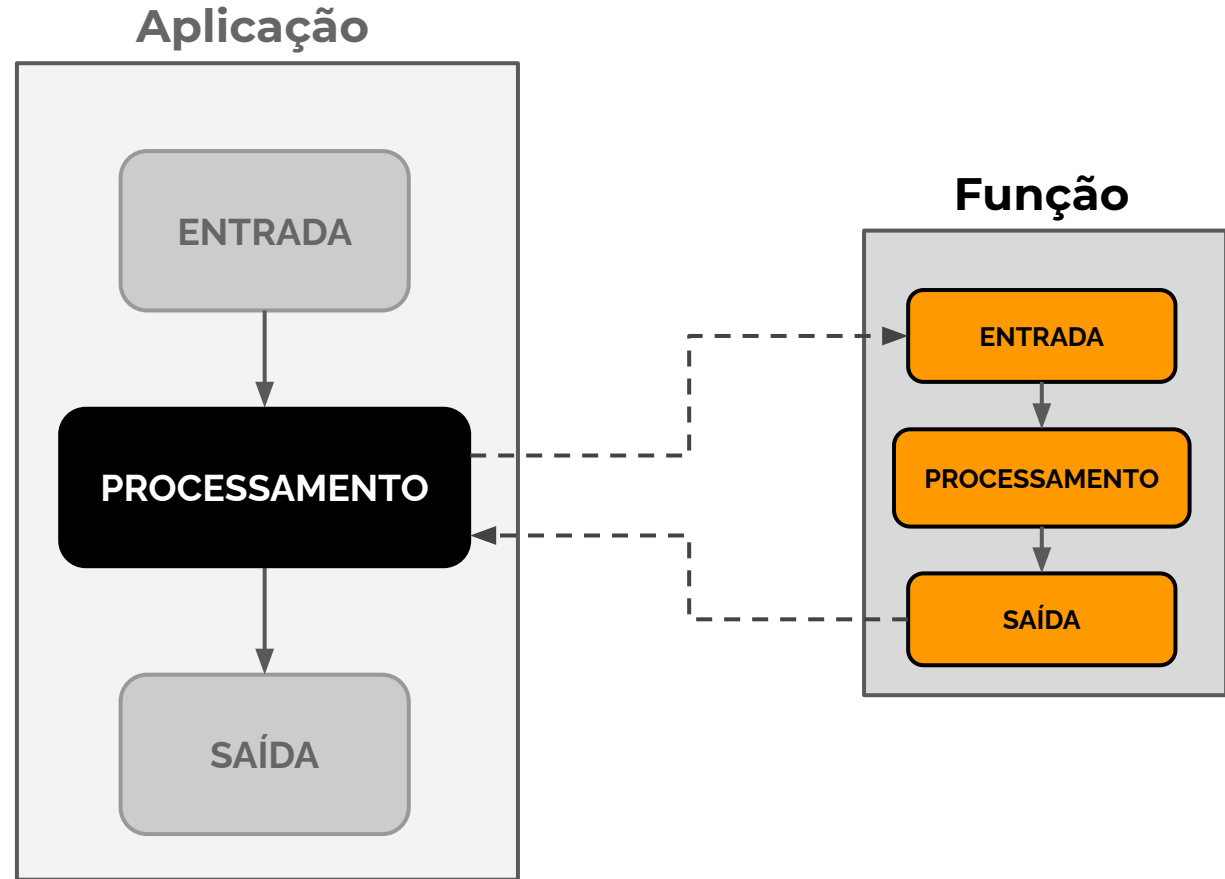
# Modelo Entrada-Processamento-Saída

## — Níveis de Abstração



# Modelo Entrada-Processamento-Saída

## Níveis de Abstração



# Modelo Entrada-Processamento-Saída

Diferentes formas

## Entrada

Console

E1

Literal

(*hard coding*)

E2

Argumentos na  
execução

E3

Arquivo

E4

## Processamento

Programa executável

## Saída

S1

Console

S2

Arquivo



# Modelo Entrada-Processamento-Saída

Diferentes formas

## Entrada

Console

E1

Literal

(*hard coding*)

E2

Argumentos na  
execução

E3

Arquivo

E4

## Processamento

```
int soma = a + b
```

## Saída

S1

Console

S2

Arquivo

# Modelo Entrada-Processamento-Saída

Diferentes formas

## Entrada

Console

E1

Literal  
(hard coding)

E2

Argumentos na  
execução

E3

Arquivo

E4

## Processamento

```
#include <stdio.h>

int main(){
    int a, b;

    // ENTRADA
    scanf("%d %d", &a, &b);

    // PROCESSAMENTO
    int soma = a + b;

    // SAÍDA
    printf("%d\n", soma);

    return 0;
}
```

## Saída

S1

Console

S2

Arquivo

# Modelo Entrada-Processamento-Saída

## Diferentes formas

### Entrada

Console

E1

**Literal**  
*(hard coding)*

E2

Argumentos na  
execução

E3

Arquivo

E4

### Processamento

```
#include <stdio.h>

int main(){
    int a, b;

    // ENTRADA
    a = 50;
    b = 50;

    // PROCESSAMENTO
    int soma = a + b;

    // SAÍDA
    printf("%d\n", soma);

    return 0;
}
```

### Saída

S1

Console

S2

Arquivo

# Modelo Entrada-Processamento-Saída

## Diferentes formas

### Entrada

Console

E1

Literal

(hard coding)

E2

Argumentos na  
execução

E3

Arquivo

E4

### Terminal

```
> gcc custo.c -o custo
> ./custo 10 20
```

### Processamento

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char const *argv[]){
    int a, b;

    // ENTRADA
    a = atoi(argv[1]);
    b = atoi(argv[2]);

    // PROCESSAMENTO
    int soma = a + b;

    // SAÍDA
    printf("%d\n", soma);

    return 0;
}
```

### Saída

S1

Console

S2

Arquivo

# Modelo Entrada-Processamento-Saída

## Diferentes formas

### Entrada

Console

E1

Literal

(hard coding)

E2

Argumentos na  
execução

E3

Arquivo

E4

entrada.in

10 20

### Processamento

```
#include <stdio.h>

int main(){
    int a, b;

    // ENTRADA
    FILE *arq_in = fopen("entrada.in", "r");
    fscanf(arq_in, "%d %d", &a, &b);
    fclose(arq_in);

    // PROCESSAMENTO
    int soma = a + b;

    // SAÍDA
    printf("%d\n", soma);

    return 0;
}
```

### Saída

S1

Console

S2

Arquivo

# Modelo Entrada-Processamento-Saída

Diferentes formas

## Entrada

Console

E1

Literal

(hard coding)

E2

Argumentos na  
execução

E3

Arquivo

E4

entrada.in

10 20

## Processamento

```
#include <stdio.h>

int main(){
    int a, b;

    // ENTRADA
    FILE *arq_in = fopen("entrada.in", "r");
    fscanf(arq_in, "%d %d", &a, &b);
    fclose(arq_in);

    // PROCESSAMENTO
    int soma = a + b;

    // SAÍDA
    FILE *arq_out = fopen("saida.out", "w");
    fprintf(arq_out, "%d\n", soma);
    fclose(arq_out);
    return 0;
}
```

## Saída

S1

Console

S2

Arquivo

saida.out

30

# Redirecionando as streams padrão para arquivos

## Usando a função freopen

### Entrada

Arquivo

E4

entrada.in

10 20

### Processamento

```
#include <stdio.h>

int main(){
    int a, b;
    // Redireciona as streams padrao
    freopen("entrada.in", "r", stdin);
    freopen("saida.out", "w", stdout);

    // ENTRADA
    scanf("%d %d", &a, &b);

    // PROCESSAMENTO
    int soma = a + b;

    // SAÍDA
    printf("%d\n", soma);

    return 0;
}
```

### Saída

S2

Arquivo

saida.out

30

# Redirecionando as streams padrão para arquivos

Utilizando argumentos na linha de comando

## Entrada

Arquivo **E4**

entrada.in

10 20

<

redireciona o stdin

>

redireciona o stdout

## Terminal

```
> gcc custo.c -o custo
> ./custo < entrada.in > saida.out
```

## Processamento

```
#include <stdio.h>

int main(){
    int a, b;
    // Redireciona as streams padrao
    freopen("entrada.in", "r", stdin);
    freopen("saida.out", "w", stdout);

    // ENTRADA
    scanf("%d %d", &a, &b);

    // PROCESSAMENTO
    int soma = a + b;

    // SAÍDA
    printf("%d\n", soma);

    return 0;
```

## Saída

**S2** Arquivo

saida.out

30



*Fim*