

Algoritmia e Estruturas de Dados

Módulo II- Strings



Strings

- **Strings**
 - ❑ Cadeia de caracteres, consiste numa sequência de símbolos alfanuméricos. Pode incluir caracteres alfabéticos, dígitos, caracteres de controle, etc.
 - ❑ Uma string sem quaisquer caracteres consiste numa string nula

Exemplo:

Algoritmo

Texto nome

nome ← " "

C#

string nome = "";

Strings

- **Concatenação de Strings**

- ❑ É uma operação sobre strings, que pode ser vista como uma operação análoga a adição na matemática

Algoritmo

Texto nome, apelido, nome_completo

nome ← "Carlos"

apelido ← "Fonseca"

nome_completo ← nome + " " + apelido

C#

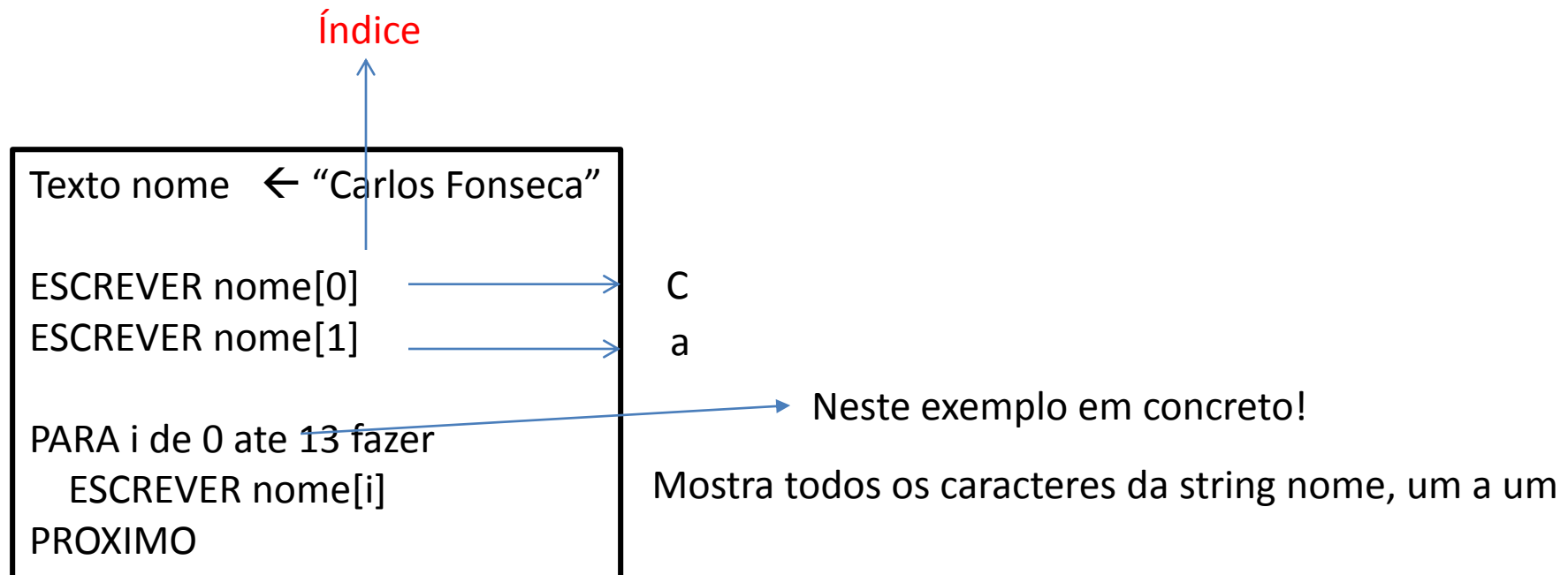
```
String nome = "Carlos";  
String apelido = "Fonseca";  
String nome_completo = nome + " " + apelido;  
Console.WriteLine(nome_completo);
```

Strings

- Índice de uma String

- ❑ Uma string consiste num conjunto de caracteres, **acessíveis individualmente** através da especificação do seu índice (posição) na string.

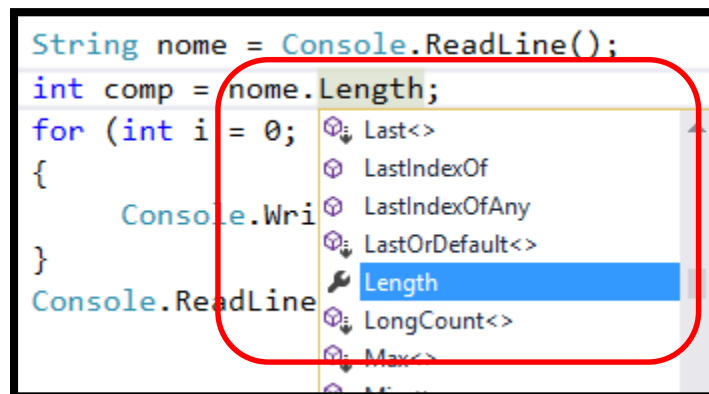
O primeiro caracter de uma string tem a posição 0.



Strings

- **Propriedades e Métodos da classe String**

- ❑ A manipulação de variáveis do tipo string oferece um vasto recurso de propriedades e métodos (funções), no C#.
- ❑ No editor do C# basta escrever o nome da variável (declarada como string) seguido de um . para que surja uma lista de propriedades e métodos que podemos invocar.



Strings

- Propriedades da classe String

Propriedades	Descrição
Length	Devolve o comprimento da string (número de caracteres)

- Length

```
string nome = "João Moutinho";  
int comprimento = nome.Length; // número de caracteres da string nome
```



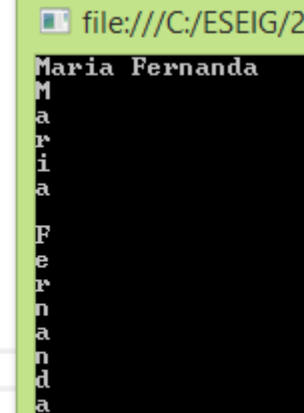
Comprimento = 13

Strings

- **Strings**

Propriedade Length: Nº de caracteres de uma string, comprimento da string

```
static void Main(string[] args)
{
    String nome = Console.ReadLine();
    int comp = nome.Length;
    for (int i = 0; i < comp; i++)
    {
        Console.WriteLine(nome[i]);
    }
    Console.ReadLine();
}
```



file:///C:/ESEIG/2

Maria Fernanda

M
a
r
i
a
F
e
r
n
a
n
d
a

Strings

- Métodos da classe String (alguns meros exemplos...)

Métodos	Descrição
ToLower()	Converte os caracteres da string para minúsculas
ToUpper()	Converte os caracteres da string para maiúsculas
IndexOf(padrão)	Devolve a posição em que se encontra, na string, o padrão de procura (primeira ocorrência)
LastIndexOf(padrão)	Devolve a posição em que se encontra, na string, o padrão de procura (última ocorrência)
Substring(posição, num_caract)	Obtem uma string com num_caracteres, a partir de posição
Replace(substring1, substring2)	Substitui <i>substring1</i> por <i>substring2</i>
Insert(posicao, string)	Insere caracteres numa string, a partir de posição
Remove(posição, num_caract)	Remove num_caract caracteres a partir de posição
StartsWith(padrão)	Verifica se a string se inicia com os caracteres padrão; se sim devolve <i>true</i> , caso contrário devolve <i>false</i>

Strings

- ToLower()

```
string nome = "João Moutinho";
int comprimento = nome.Length;           // número de caracteres da string nome

string nome1 = nome.ToLower();           // string nome1 contém os caracteres da string nome,
                                         // convertidos em minúsculas

string nome2 = nome.ToUpper();           // string nome2 contém os caracteres da string nome,
                                         // convertidos em maiúsculas
```

- ToUpper()

joão moutinho

JOÃO MOUTINHO

- IndexOf(*padrao_procura*)

```
string nome = "João Moutinho";
int comprimento = nome.Length;           // número de caracteres da string nome

int posicao = nome.IndexOf(" ");          // Procura na string nome a posicao em que se encontra, neste caso, o caracter " "
                                         // Se existirem diversos espaços, a função IndexOf devolve a primeira ocorrencia
```

Posicao = 4

Strings

- `LastIndexOf(padrao_procura)`

```
string nome = "João Moutinho";  
int comprimento = nome.Length;    // número de caracteres da string nome  
  
int posição = nome.IndexOf(" ");  
  
int pos1 = nome.LastIndexOf("h");  // procura na string nome a última ocorrência do caracter h
```

- `Substring(posição, nº de caracteres)`

```
string nome = "João Moutinho";  
int comprimento = nome.Length;    // número de caracteres da string nome  
  
int posicao = nome.IndexOf(" ");    // Procura na string nome a posicao em que se encontra, neste caso, o caracter " "  
                                   // Se existirem diversos espaços, a função IndexOf devolve a primeira ocorrencia  
  
string nome_proprio = nome.Substring(0, posicao);  
string apelido = nome.Substring(posicao+1, comprimento - posicao-1);  
  
Console.WriteLine(nome_proprio);   → João  
Console.WriteLine(apelido);        → Moutinho  
Console.ReadLine();
```

Strings

- *Replace(substring1, substring2)*

João Pinto ←

```
string nome = "João Moutinho";  
  
nome = nome.Replace("Moutinho", "Pinto"); // Substitui uma substring por outra substring  
Console.WriteLine(nome);
```

```
// os argumentos da função Replace podem ser variáveis  
string variavel2 = "Pereira";  
nome = nome.Replace("Pinto", variavel2);
```

João Pereira ←

```
Console.WriteLine(nome);  
Console.ReadLine();
```

Strings

- *Insert(posicao, substring)*

```
string nome = "João Moutinho";

nome = nome.Insert(5, "Manuel ");

Console.WriteLine(nome);
Console.ReadLine();
```



João Manuel Moutinho

```
string nome = "João Moutinho";

// nome = nome.Insert(5, "Manuel ");

// Exemplo usando variáveis *****
int posicao = nome.IndexOf(" ");
string var2 = "Manuel ";
nome = nome.Insert(posicao+1, var2);

Console.WriteLine(nome);
Console.ReadLine();
```

- *Remove(posicao, substring)*

```
string nome = "João Manuel Moutinho";

int pos = nome.IndexOf("Manuel"); // devolve a primeira posição de Manuel
nome = nome.Remove(pos, 7);

|
Console.WriteLine(nome);
Console.ReadLine();
```

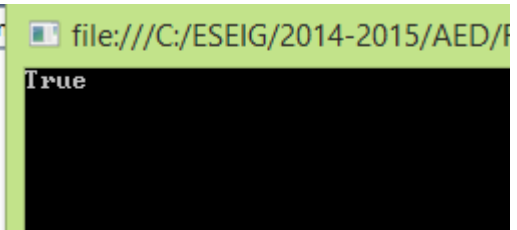


João Moutinho

Strings

- *StartsWith(padrão de pesquisa)*

```
string nome = "João Moutinho";  
  
bool inicia = nome.StartsWith("João");  
Console.WriteLine(inicia);  
  
Console.ReadKey();
```



file:///C:/ESEIG/2014-2015/AED/f
True

Devolve *True* de a string se inicia pelo padrão de pesquisa,
Caso contrário devolve *False*

Strings

Remover espaços iniciais e/ finais numa string

- *TrimStart()* // remove os espaços iniciais à esquerda
- *TrimEnd* // remove os espaços à direita do último carácter
- *Trim()* // remove os espaços que possam existir à esquerda do primeiro carácter e à direita do último

```
string nome = "    João Manuel Moutinho    ";  
string nome1 = nome.TrimStart();  
Console.WriteLine(nome1);  
  
nome = "    João Manuel Moutinho    ";  
nome1 = nome.TrimEnd();  
Console.WriteLine(nome1);  
  
nome = "    João Manuel Moutinho    ";  
nome1 = nome.Trim();  
Console.WriteLine(nome1);
```

```
João Manuel Moutinho  
João Manuel Moutinho  
João Manuel Moutinho
```