

Aula 10

Strings (cadeias de caracteres)

Programação em Java 2006-2007

Declaração e criação

```
String nome_do_string = new String("texto")
```

Exemplo:

```
String str = new String("Bom dia turma");
```

Também podemos escrever mais directamente:

```
String nome_do_string = "texto"
```

Exemplo:

```
String str = "Bom dia turma";
```

Os strings são constantes: uma vez criado um string com determinado texto este não pode ser modificado.

Programação em Java 2006-2007

2

A função `length`

Para obter o número de caracteres de um `String` usamos a função `length()`

Exemplo:

```
//cria um string com nome str
String str = "Bom dia turma";
//imprime no ecrã o número de caracteres do string str
print("O número de caracteres é " + str.length());
```

Será impresso no ecrã:
O número de caracteres é 13.

Note que como
é uma função
devemos por ()

Programação em Java 2006-2007

3

A função `charAt`

O acesso a cada um dos caracteres de um `String` é feito através da função `charAt()`.

O primeiro carácter tem o índice **zero**, enquanto que o último tem o índice igual à `length() - 1`

Exemplo:

```
String str = "Bom dia turma";
//imprime no ecrã o décimo carácter do string str
println("O décimo carácter é " + str.charAt(9));
```

Será impresso no ecrã:
O décimo carácter é u.

0	1	2	3	4	5	6	7	8	9	10	11	12
B	o	m		d	i	a		t	u	r	m	a

`str.charAt(0)` `str.charAt(9)` `str.charAt(str.length()-1)`

Programação em Java 2006-2007

4

Problema 1

```
import p1.*;
class Pr_10_1_Strings extends P1App {
    static void main(String [] args) {
        String s = new String();
        char c;
        int i, nEsp = 0;
        // ler uma linha
        s = readLine("Escreva uma linha de texto: ");
        // mostrar o tamanho da linha em caracteres
        println("A linha tem " + s.length() + " caracteres, sendo.");
        // mostrar o nº de espaços
        for (i=0; i<s.length(); i++) {
            c = s.charAt(i);
            if (c == ' ')
                nEsp++;
        }
        println(" . " + nEsp + " espaços.")
    }
}
```

Completar o programa por forma que mostre no ecrã o número de:

- ✓ letras minúsculas
- ✓ letras maiúsculas,
- ✓ dígitos
- ✓ outros caracteres

A função substring

Podemos extrair um subcadeia de caracteres de um string dado usando a função **substring**

Exemplo:

```
String str = "Bom dia turma";
```

```
//imprime uma subcadeia de str a partir do índice 8
```

```
println (str.substring(8));
```

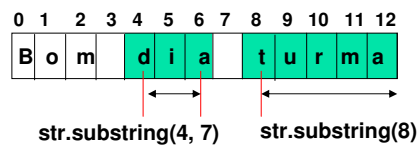
Será impresso no ecrã:
turma

```
//imprime uma subcadeia de str desde o índice 4 até o 6
```

```
println (str.substring(4, 7));
```

Será impresso no ecrã:
dia

(aqui devemos indicar o índice do último caracter + 1, por isso fica 7)



toLowerCase e toUpperCase

Podemos converter todos os caracteres de um string em minúsculas ou maiúsculas usando as funções **toLowerCase** e **toUpperCase**

Exemplo:

```
String str = "Bom dia turma";  
String minStr = str.toLowerCase();  
String maiuStr = str.toUpperCase();  
println (minStr);  
println (maiuStr);
```

Cria um novo String chamado **minStr** com o texto armazenado em **str** todo com caracteres minúsculos

Será impresso no ecrã:
bom dia turma

Será impresso no ecrã:
BOM DIA TURMA

Programação em Java 2006-2007

A função indexOf

Podemos determinar a posição de um carácter dado num string usando a função **indexOf**

Exemplo:

```
String str = "Bom dia turma";  
// retorna em pos o índice onde se encontra a primeira "a"  
int pos = str.indexOf('a');  
// retorna em pos o índice onde se encontra a seguinte "a"  
pos = str.indexOf('a', pos+1);
```

pos toma o valor **6**

procura a partir do índice **7 (pos +1)**.
pos agora toma o valor **12**,

```
str.indexOf('a')
```

0	1	2	3	4	5	6	7	8	9	10	11	12
B	o	m		d	i	a		t	u	r	m	a

Programação em Java 2006-2007

A função `indexOf`

Podemos também usar `indexOf` para determinar a posição de um substring dado

Exemplo:

```
String str = "Bom dia turma";
```

```
// retorna em pos o índice onde se encontra a palavra "dia"
```

```
int pos = str.indexOf("dia");
```

pos toma o valor 4

```
pos = str.indexOf("colega");
```

procura o substring "colega". Como não está em str retorna o valor -1

```
str.indexOf("dia")
```

0	1	2	3	4	5	6	7	8	9	10	11	12
B	o	m		d	i	a		t	u	r	m	a

Programação em Java 2006-2007

9

A função `compareTo`

Compara um string (`str1`) com um outro string dado (`str2`) e retorna um valor inteiro:

< 0, se `str1` é menor lexicograficamente do que `str2` (`str1 << str2`)

= 0, se `str1` é igual a `str2`

> 0, se `str1` é maior lexicograficamente do que `str2` (`str1 >> str2`)

Exemplo:

```
String nome1 = "Marcelo";
```

```
String nome2 = "Antonio";
```

res > 0 pois Marcelo >> Antonio

```
int res = nome1.compareTo(nome2);
```

```
res = nome2.compareTo(nome1);
```

res < 0 pois Antonio << Marcelo

Programação em Java 2006-2007

10

A função `compareToIgnoreCase`

Similar à função `compareTo` mas esta não diferencia as letras maiúsculas das minúsculas.

Exemplo:

```
String nome1 = "MARCELO";  
String nome2 = "antonio";  
int res = nome1.compareTo(nome2);  
res = nome1.compareToIgnoreCase(nome2);
```

res < 0 pois MARCELO << antonio

res > 0 pois MARCELO >> antonio
se não considerarmos diferentes
maiúsculas e minúsculas

A função `concat`

`nome_string_1.concat(nome_string_2)` ↔ `nome_string_1 + nome_string_2`

Exemplo:

```
String str1 = "Só sei ";  
String str2 = "que nada sei";  
String str = str1.concat(str2);  
println(str);
```

```
String str1 = "Só sei ";  
String str2 = "que nada sei";  
String str = str1 + str2;  
println(str);
```

Em ambos casos será impresso no ecrã:
Só sei que nada sei



Problema 2

- Escrever um programa que lê o endereço email de um aluno da UAveiro e escreve no ecrã o nº mecanográfico, a sigla do departamento e o nome do departamento correspondentes. Considerar apenas as siglas egi e det. A interacção será da seguinte forma:

```
Endereço email      : a9876@egi.ua.pt
Nº mecanográfico    : 9876
Sigla do departamento : EGI
Nome do Departamento : Economia e Gestão Industrial
```

Nota: a sigla do departamento deve ser escrita em letras maiúsculas.



Bibliografia

- António José Mendes, Maria José Marcelino. *Fundamentos de programação em JAVA 2.FCA* – Editora de informática, 2003.
- Tutorial de Java
<http://java.sun.com/docs/books/tutorial/java/data/manipstrings.html>