

Sétima Lista de Exercícios

Strings e E/S

Norton Trevisan Roman

17 de maio de 2011

1. Escreva um método que retorne o tamanho de um string passado por parâmetro (não use `length`).
2. Escreva um método que implemente o método `equals` da classe `String`.
3. Escreva um método que, dado um string e um caracter passados como parâmetros, retorne a posição da primeira ocorrência do caracter no string, ou -1 caso ele não esteja lá. Não use `indexOf` da classe `String`.
4. Escreva um método que, dado um string, imprima a primeira letra após cada espaço. Ou seja, ele deve fazer algo assim:

```
Minha string é assim.  
s é a
```

5. Um string é palíndromo se ele é igual tanto lido da esquerda para a direita, quanto da direita para a esquerda. Por exemplo, "ovo" e "osso" são palíndromos. Escreva um método que determine se um string é palíndromo ou não.
6. Faça um método que receba um string como parâmetro e o inverta.
7. Escreva um método que crie uma cópia de um string. O método deve receber o string como parâmetro e devolver uma cópia (em memória, ou seja, um objeto diferente) dele.
8. Escreva um método que receba como parâmetro uma linha de caracteres e, a partir de tal linha, lendo um caractere por vez, produza as subsequências contíguas de caracteres não brancos, em um arranjo, uma por posição no arranjo, retornando esse arranjo. Exemplo: para a entrada

```
as12;  bbcdd  ---  987  .:.:.  @Alf@#
```

o método deve retornar um arranjo contendo:

```
as12;  
bbcdd  
---  
987  
.:.:.  
@Alf@#
```

9. Escreva um programa que leia um string do usuário e o converta em um número float. O string deve estar seguindo os padrões brasileiros. Exemplos de entrada:

1234,56
1.234,56
1.234.654,89
1234654,89

10. Escreva um programa que leia primeiramente um valor $n \geq 0$ do usuário e, a seguir, três caracteres por linha, nas n linhas que se seguem. Para cada grupo de três caracteres o programa deve informar se o grupo está com três caracteres iguais, ordenados em ordem crescente, ordenados em ordem decrescente ou desordenados.
11. Escrever um programa que, enquanto as respostas do usuário forem positivas, repita o seguinte procedimento: consultar ao usuário se ele quer efetuar o cálculo do valor absoluto de um valor real e, em caso afirmativo, ler um valor real e apresentar o valor absoluto de tal valor. Em caso negativo, a execução do programa deve ser encerrada.
12. Escreva um programa que leia uma frase de 20 caracteres do usuário e
- (a) conte o número de espaços em branco
 - (b) conte o número de letras 'a'.
13. Escrever um método que determine o valor mínimo e máximo de um vetor de n valores fornecidos pelo usuário.
14. Escreva uma classe que calcule a média de N números dados pelo usuário, onde N não é conhecido. A cada iteração o usuário responde se quer continuar ou não. Toda a interação com o usuário se dá via método *main*. O método para cálculo da média deve receber um arranjo com os n números digitados pelo usuário.
15. Escreva uma classe que apresente os n primeiros números primos a partir do número 2 para um valor $n > 0$ fornecido pelo usuário.
16. Escreva um programa que calcule a nota de aproveitamento (A) de um aluno de acordo com o critério de avaliação da nossa disciplina. O usuário do programa deve fornecer as notas obtidas por um aluno nas provas e a seguidas pelas notas obtidas em atividades de laboratório.
17. Escreva um método que verifique se um número inteiro positivo é potência de 2. Em caso afirmativo, o método deve fornecer o valor da potência. No programa principal devem ser lidos vários números até que o usuário indique que não deseja repetir a verificação para nenhum outro número.
18. Escreva um programa que leia do usuário uma lista de inteiros positivos, terminada por -1 (que deve ser usado apenas para sinalizar o fim da lista) e imprima esta lista em ordem inversa. Use uma variável para definir o número máximo de inteiros que o programa poderá ler (atenção! É o número máximo, você pode ler menos que isso). O programa deve dar uma mensagem se o número de inteiros lidos ultrapassar esse máximo, indicando que os adicionais serão ignorados.
19. Escreva um programa que leia uma string digitada pelo usuário e coloque todas as primeiras letras de cada palavra nesse string em maiúscula. Assim, se a entrada for “a casa azul” a saída será “A Casa Azul”.

20. Escrever um programa que aponte para os dígitos (se existirem) em uma linha de texto fornecida pelo usuário. Isto é, o programa deve gerar uma linha logo abaixo da linha fornecida pelo usuário que, em cada posição equivalente a um dígito na linha original, tenha um caractere '^' e um espaço nas demais. Exemplo:

Mariazinha comprou 3 goiabas, 1 melancia e 15 bananas na feira.
 ^ ^ ^^

21. Escrever um programa que deve ler pares de valores de uma linha de entrada, onde cada par é composto por um valor inteiro representando unidades e um valor real representando um preço unitário de um produto não especificado. A partir de tais valores o programa deve montar uma tabela de três colunas alinhadas (imagine a tabela como uma nota fiscal super simplificada), em que cada linha temos as unidades, o preço unitário e o valor total referente ao produto a que se refere o par correspondente na linha de entrada. Ao final da tabela deve ser gerada uma linha em que é colocado o valor da soma dos valores da terceira coluna da tabela. O total geral deve estar alinhado com os valores na terceira coluna. Exemplo: para a linha de entrada

3 15.20
1 8.35
5 3.70
4 12.00

o programa deve gerar algo como

3	15.20	45.60
1	8.35	8.35
5	3.70	18.50
4	12.00	48.00
		120.45

22. Implemente uma classe com um método que determina a frequência relativa de vogais em uma String.