

Análise e Desenvolvimento de Sistemas - ADS

Linguagem de Programação - LP

LISTA 03 DE EXERCÍCIOS

Prof. Cristóvão Cunha

Assunto: Operadores

Esta lista contém 14 exercícios que devem ser entregues ao professor, resolvidos em linguagem de programação Java, dentro de uma pasta com o Seu Nome Completo, não sendo aceito o envio pela Internet. Dentro de cada exercício (arquivo fonte) deve haver um comentário com o Seu Nome Completo. Estes exercícios podem ser feitos em casa ou na monitoria, com orientação do monitor ou com a ajuda do professor.

1 - Crie um arquivo chamado ConversaoPrimitivos.java. Para eliminar os erros de compilação, complete os espaços marcados por “xxxxx” do código abaixo quando necessário.

```
1  public class ConversaoPrimitivos {
2      public static void main(String[] args) {
3          int a      = 10;
4          long b      = xxxxx a;
5          int c       = xxxxx b;
6          float d     = xxxxx b;
7          short e     = xxxxx a;
8          double f    = xxxxx d;
9          Long g      = xxxxx b;
10         long h      = xxxxx g;
11
12         System.out.println(a);
13         System.out.println(b);
14         System.out.println(c);
15         System.out.println(d);
16         System.out.println(e);
17         System.out.println(f);
18         System.out.println(g);
19         System.out.println(h);
20     }
21 }
```

2 - Crie um arquivo chamado Casting.java. Crie uma variável chamada numero1 do tipo long e inicialize-a com o valor 2147483648L. Depois, crie uma variável chamada numero2 do tipo int e copie o valor da variável numero1 para a variável numero2. Por fim, exiba, na saída padrão, os valores dessas duas variáveis.

3 - Crie um arquivo chamado Conversao.java. Crie uma variável chamada texto do tipo String e associe a ela a sequência de caracteres "19.09". Depois, faça a conversão do valor associado a essa variável para double e armazene o resultado em uma variável chamada numero do tipo double. Por fim, exiba o valor da variável numero.

4 - Crie um arquivo chamado OperadoresAritmeticos.java. Complete o código a seguir no local indicado por "xxxxx", com os operadores aritméticos da linguagem Java. Não utilize operadores repetidos. O programa deve exibir os números 11, 8, 60, 5 e 2 nessa ordem.

```
1 public class OperadoresAritmeticos {
2     public static void main(String[] args) {
3         int a = 10 xxxxx 1;
4         int b = 10 xxxxx 2;
5         int c = 20 xxxxx 3;
6         int d = 25 xxxxx 5;
7         int e = 10 xxxxx 4;
8
9         System.out.println(a);
10        System.out.println(b);
11        System.out.println(c);
12        System.out.println(d);
13        System.out.println(e);
14    }
15 }
```

5 - Crie um arquivo chamado TiposDeResultado.java. Complete o código a seguir, substituindo a sequência "xxxxx" com os tipos primitivos numéricos da linguagem Java. Utilize sempre o tipo que ocupa o menor espaço na memória.

```
1 public class TiposDeResultado {
2     public static void main(String[] args) {
3         byte a = 1;
4         short b = 2;
5         double c = 3.14;
6
7         xxxxx d = a + b;
8         xxxxx e = b + c;
9
10        System.out.println(d);
11        System.out.println(e);
12    }
13 }
```

6 - Crie um arquivo chamado DivisaoInteiraReal.java. Complete o código, substituindo a sequência "xxxxx" com operações de divisão, operações de casting e parênteses. O programa deve exibir os números 2, 2.5, 2.5, 2.5 e 2.0 nessa ordem. A cada linha complete o código de uma forma diferente. Não é necessário completar todas as sequências.

```
1 public class DivisaoInteiraReal {
2     public static void main(String[] args) {
3         int a = 5;
4         int b = 2;
5
6         System.out.println( xxxxx a xxxxx b xxxxx );
7         System.out.println( xxxxx a xxxxx b xxxxx );
8         System.out.println( xxxxx a xxxxx b xxxxx );
9         System.out.println( xxxxx a xxxxx b xxxxx );
10        System.out.println( xxxxx a xxxxx b xxxxx );
11    }
12 }
```

7 - Crie um arquivo chamado OverflowUnderflow.java. Complete o código, substituindo a sequência “xxxxx” com valores literais. O programa deve exibir os números -2147483648 e 2147483647 nessa ordem.

```
1 public class OverflowUnderflow {
2     public static void main(String[] args) {
3         System.out.println( xxxxx + 1 );
4         System.out.println( xxxxx - 1 );
5     }
6 }
```

8 - Crie um arquivo chamado Concatenacao.java. Complete o código, substituindo a sequência “xxxxx” com operações de concatenação e parênteses. O programa deve exibir as mensagens "Java123", "Java6", "123Java" e "6Java" nessa ordem. Não é necessário completar todas as sequências.

```
1 public class Concatenacao {
2     public static void main(String[] args) {
3         System.out.println( xxxxx "Java" xxxxx 1 xxxxx 2 xxxxx 3 xxxxx );
4         System.out.println( xxxxx "Java" xxxxx 1 xxxxx 2 xxxxx 3 xxxxx );
5         System.out.println( xxxxx 1 xxxxx 2 xxxxx 3 xxxxx "Java" xxxxx );
6         System.out.println( xxxxx 1 xxxxx 2 xxxxx 3 xxxxx "Java" xxxxx );
7     }
8 }
```

9 - Crie um arquivo chamado OperadoresDeAtribuicao.java. Complete o código, substituindo a sequência “xxxxx” com operações de atribuição. O programa deve exibir os valores 1, 3, 2, 6, 3, 1, 2 e 1 nessa ordem. Não utilize o mesmo operador duas ou mais vezes.

```
1 public class OperadoresDeAtribuicao {
2     public static void main(String[] args) {
3         int a xxxxx 1;
4         System.out.println(a);
5
6         a xxxxx 2;
7         System.out.println(a);
8 }
```

```
9      a xxxxx 1;
10     System.out.println(a);
11
12     a xxxxx 3;
13     System.out.println(a);
14
15     a xxxxx 2;
16     System.out.println(a);
17
18     a xxxxx 2;
19     System.out.println(a);
20
21     a xxxxx;
22     System.out.println(a);
23
24     a xxxxx;
25     System.out.println(a);
26 }
27 }
```

10 - Crie um arquivo chamado OperadoresRelacionais.java. Complete o código, substituindo a sequência “xxxxx” com operações de comparação. O programa deve exibir os valores false, false, true, true, false e true nessa ordem. Não utilize o mesmo operador duas ou mais vezes.

```
1  public class OperadoresRelacionais {
2      public static void main(String[] args) {
3          int a = 1;
4          int b = 2;
5
6          System.out.println(a xxxxx b);
7          System.out.println(a xxxxx b);
8          System.out.println(a xxxxx b);
9          System.out.println(a xxxxx b);
10         System.out.println(a xxxxx b);
11         System.out.println(a xxxxx b);
12     }
13 }
```

11 - crie um arquivo chamado OperadoresLogicos.java. Complete o código, substituindo a sequência “xxxxx” com operações lógicas. O programa deve exibir os valores true, true, false, false e true nessa ordem. Não utilize o mesmo operador duas ou mais vezes.

```
1  public class OperadoresLogicos {
2      public static void main(String[] args) {
3          int a = 1;
4          int b = 2;
5          int c = 3;
6          int d = 4;
7
8          System.out.println(a > b xxxxx c < d);
9          System.out.println(a > b xxxxx c < d);
10         System.out.println(a > b xxxxx c < d);
11         System.out.println(a > b xxxxx c < d);
12         System.out.println(a > b xxxxx c < d);
13     }
14 }
```

12 - crie um arquivo chamado OperadorTernario.java. Complete o código, substituindo a sequência “xxxxx” com o operador ternário. O programa deve exibir a mensagem $a < b$ quando o valor da variável a for menor do que o valor da variável b e $a \geq b$ caso contrário.

```
1 public class OperadorTernario {
2     public static void main(String[] args) {
3         int a = (int)(Math.random() * 100);
4         int b = (int)(Math.random() * 100);
5
6         System.out.println(a < b xxxxx );
7     }
8 }
```

13 - Crie um arquivo chamado OperadorNegacao.java. Complete o código, substituindo a sequência “xxxxx” com o operador de negação e parênteses. O programa deve exibir os valores true e false nessa ordem. Não é necessário preencher todas as sequências.

```
1 class OperadorNegacao {
2     public static void main(String[] args) {
3         int a = 10;
4         int b = 20;
5
6         System.out.println( xxxxx a < b xxxxx );
7         System.out.println( xxxxx a < b xxxxx );
8     }
9 }
```

14 - Crie um arquivo chamado PrePosIncrementoDecremento.java. Complete o código, substituindo a sequência “xxxxx” com os operadores de incremento e decremento. O programa deve exibir os valores 1, 3, 3 e 1 nessa ordem. Não é necessário preencher todas as sequências.

```
1 class PrePosIncrementoDecremento {
2     public static void main(String[] args) {
3         int a = 1;
4
5         System.out.println( xxxxx a xxxxx );
6         System.out.println( xxxxx a xxxxx );
7         System.out.println( xxxxx a xxxxx );
8         System.out.println( xxxxx a xxxxx );
9     }
10 }
```