

MASTERING JAVA PROGRAMS



**FIREWALL
MEDIA**

J.B. Dixit

Published by :

FIREWALL MEDIA

(An Imprint of Laxmi Publications Pvt. Ltd.)

113, Golden House, Daryaganj,
New Delhi-110002

Phone : 011-43 53 25 00

Fax : 011-43 53 25 28

www.laxmipublications.com
info@laxmipublications.com

*Copyright © 2010 by Laxmi Publications Pvt. Ltd. All rights reserved.
No part of this publication may be reproduced, stored in a retrieval system, or
transmitted in any form or by any means, electronic, mechanical, photocopying,
recording or otherwise without the prior written permission of the publisher.*

Edited by: SANGEETA DIXIT

Price : Rs. 250.00 Only.

First Edition : 2010

OFFICES

© Bangalore	080-26 61 15 61	© Chennai	044-24 34 47 26
© Cochin	0484-237 70 04, 405 13 03	© Guwahati	0361-254 36 69, 251 38 81
© Hyderabad	040-24 65 23 33	© Jalandhar	0181-222 12 72
© Kolkata	033-22 27 43 84	© Lucknow	0522-220 95 78
© Mumbai	022-24 91 54 15, 24 92 78 69	© Ranchi	0651-221 47 64

CONTENTS

Preface ... (xvii)

Compiling and Running a Java Program ... (xix)

Chapters *pages*

1. Elementary Programs ... **1—100**

<u>1. Illustration of basic arithmetic operators</u>	...	1
<u>2. Illustration of the bitwise logical operators</u>	...	3
<u>3. Illustration of Boolean logical operators</u>	...	4
<u>4. Use of comma operator</u>	...	5
<u>5. Swap (interchange) two number using a temporary variable</u>	...	6
<u>6. Swap (interchange) two numbers without using a temporary variable</u>	...	7
<u>7. Find simple interest</u>	...	8
<u>8. Compute surface area and volume of a cube</u>	...	9
<u>9. Convert given number in years, weeks and days (ignore leap year)</u>	...	10
<u>10. Calculate the sum and average of five numbers</u>	...	11
<u>11. Use of conditional operator</u>	...	12
<u>12. Leap year checking</u>	...	13
<u>13. Find smallest of three numbers</u>	...	14
<u>14. Find biggest of three numbers</u>	...	15
<u>15. Find summation of series $a + ar + ar^2 + \dots + ar^{(n-1)}$</u>	...	16
<u>16. Simulate a simple calculator</u>	...	17
<u>17. Illustration of switch..case control structure</u>		
<u>Input a number from 1–7 and write corresponding day of week</u>	...	19
<u>18. Find area of a triangle and its type</u>	...	20
<u>19. Find grade of a student</u>	...	22
<u>20. Print all possible combination of 3 digits</u>	...	24

21. Find the roots of a quadratic equation	...	26
22. Calculate the area of either circle or rectangle or triangle depending upon the user's choice	...	29
23. Print first n natural numbers and their sum using while loop	...	31
24. Calculate the occurrences of positive numbers, negative numbers and zeros in a stream of data terminated by some specific value	...	32
25. Print first n natural numbers and their sum using do-while loop	...	33
26. Print first n natural numbers and their sum using for loop	...	34
27. Multiplication table of a given number	...	35
28. Print first n natural numbers in ascending/descending order	...	36
29. Count number of 1's in the binary representation of an integer	...	37
30. Convert centigrade temperature to Fahrenheit and vice versa	...	38
31. Print number of days in a month	...	39
32. Find GCD and LCM of two non negative numbers	...	40
33. Find the number and their sum between 100 and 200 which are divisible by 7	...	41
34. Read N numbers iteratively and print biggest and smallest of these numbers	...	42
35. Find sum of even and odd numbers from 1 to n	...	43
36. Read 10 integer from the keyboard and print number of negative and positive integers	...	44
37. Check an integer for perfect square	...	45
38. Find all divisor of a positive integer	...	46
39. Sum of digits of a number and reverse it	...	47
40. Reverse a given integer and check it for palindrome	...	48
41. Print sum of first n even/odd numbers in ascending/descending order	...	49
42. Print composite number between 1 to n	...	51
43. Determine all Pythagorean triplets in the range 1 to 100. A Pythagorean triplet is a set of three integer i, j, k, such that $\sqrt{i} + \sqrt{j} = \sqrt{k}$...	53
44. Find the factorial of a number	...	56
45. Print the prime factor(s) of a positive integer	...	57
46. Check a number for perfect number	...	59
47. Print perfect numbers upto a specific limit	...	60
48. Check a numbers for Armstrong. It is a number which equals the sum of cubes of it's digits	...	61
49. Generate Armstrong numbers upto a specific limit	...	62
50. Generate first n Fibonacci terms	...	63

51. Generate Fibonacci terms upto a specific limit	...	64
52. Check a number for Fibonacci term	...	65
53. Check a number for prime. A number $p > 1$ is prime if it is not divisible by any integer from 2 to integral part of its square root	...	66
54. Generate first n prime numbers	...	67
55. Generate primes upto a specific limit and print their number	...	69
<u>56. Print the following pattern</u>	...	71

1
1 2
1 2 3
1 2 3 4
1 2 3 4 5

<u>57. Print the following pattern</u>	...	72
---	-----	----

1
2 2
3 3 3
4 4 4 4

<u>58. Print the following pattern</u>	...	73
---	-----	----

1 2 3 4 5 6
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1

<u>59. Print the following pattern</u>	...	74
---	-----	----

6 5 4 3 2 1
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1

<u>60. Print the following pattern</u>	...	75
---	-----	----

* * * * *
* * * *
* * *
* *
*

<u>61. Print the following pattern</u>	...	76
1234		
123		
12		
1		
12		
123		
1234		

62. Generate the pattern of n lines using nested loops	...	78
A		
ABC		
ABCDE		
.		
.		
.		
n lines		
63. Generate the pyramid	...	80
1		
121		
12321		
1234321		

64. Generate the pyramid using nested loops	...	82
1		
232		
34543		
4567654		
567898765		
67890109876		
7890123210987		
890123454321098		
90123456765432109		
0123456789876543210		
65. Generate the pattern of digits	...	84
66. Generate the pattern	...	86
1 = 1		
1 + 2 = 3		
1 + 2 + 3 = 6		
1 + 2 + 3 + 4 = 10		
1 + 2 + 3 + 4 + 5 = 15		
1 + 2 + 3 + 4 + 5 + 6 = 21		

67. Printing floyd triangle	...	88
1		
23		
456		
78910		
68. Sum of the series $x+(x^2)+(x^3)+(x^4)+(x^5)+\dots$...	89
69. Find the value of $\sin(x)$ using the series $x-x^3/3!+x^5/5!+\dots$ upto n terms accuracy without using the user defined function and print $\sin(x)$ using library function also	...	90
70. Find the value of $\cos(x)$ using the series $1-x^2/2!+x^4/4!+\dots$ upto n terms accuracy without using the user defined function and print $\cos(x)$ using library function also	...	92
71. Convert decimal numeral to roman numeral	...	94
72. Illustration of command line arguments	...	97
73. Illustration of reading data from keyboard using <code>readLine()</code>	...	98
74. Illustration of reading data from keyboard and calculate simple and compound interest	...	99

2. Methods

... 101—139

1. Display general message using method	...	101
2. Calculate compound interest using method	...	102
3. Add three numbers using method	...	103
4. Find square of a number using method	...	104
5. Find largest of three numbers	...	105
6. Largest of three numbers using method returning a value	...	106
7. Find the factorial of a number non recursively	...	107
8. Implement a^b where a and b are integers	...	108
9. Find the primes, their count, sum and average between 10 and 50	...	109
10. Implement the investment equation $v=p(1+r)^n$ using method without arguments	...	111
11. Illustration of nesting of methods	...	113
12. Illustration of a static variable	...	114
13. Display welcome message using method	...	115
14. Swapping of two numbers	...	116
15. Calculate the area and perimeter of a circle	...	117
16. Set smaller of two integers to 0	...	118
17. Base conversion using method	...	119
18. Sum of series $1+1/1!+1/2!+1/3!+1/4!+\dots$...	121

<u>19. Sum of the series $x - (x^2)/2! + (x^3)/3! - (x^4)/4! + \dots$</u>	...	122
<u>20. Compute sum of the series $x + x^2/3! + x^3/5! + \dots + x^{n(2n-1)}$</u>	...	123
<u>21. Compute sum of series $y + y^3/2! + y^5/3! + \dots + y^{(2m-1)}/m!$</u>	...	124
<u>22. Sum of sine series</u>	...	125
<u>23. Sum of cosine series</u>	...	126
<u>24. Convert 2-digit octal number to binary equivalent</u>	...	128
<u>25. Find divisors of a positive integer using nested method</u>	...	130
<u>26. Find standard deviation of n numbers</u>	...	132
<u>27. Display word equivalent of a number (of 1–3 digits)</u>	...	134
<u>28. Read two integer numbers and find their sum, difference, multiplication and division using a separate method for each of these operations</u>	...	138

3. Recursion ... 140—157

<u>1. Find sum of digits of a number</u>	...	140
<u>2. Factorial of a number</u>	...	142
<u>3. Find $C(n,r)$ using factorial function recursively</u>	...	143
<u>4. Find $C(n,r)$ recursively</u>	...	145
<u>5. Gcd of two positive integers (highest common factor)</u>	...	147
<u>6. Generate first n Fibonacci terms</u>	...	148
<u>7. Add two positive integers using recursion</u>	...	149
<u>8. Multiply two positive integers using recursion</u>	...	150
<u>9. Tower of hanoi</u>	...	151
<u>10. Ackermann's function</u>	...	152
<u>11. Compute the positive integer or negative integers exponential power of floating point number</u>	...	153
<u>12. Evaluate $f(x) = 0$ $x = 1$</u> <u>$f(x/2)+1$ if $x > 1$</u>	...	155
<u>13. Suppose A is an integer array with n elements and let F is an integer function defined by</u> $F(i) = F(A, n, i) = 0 \quad \text{if} \quad i = 0$ $F(i-1) + a[i] \quad \text{if} \quad 0 < i <= n$ $F(i-1) \quad \text{if} \quad i > n$...	156

4. Classes and Objects	... 158—182
<hr/>	
1. Illustration of using methods	... 158
2. Illustration of constructor	... 160
3. Application of constructors	... 161
4. Illustration of constructor overloading	... 162
5. Illustration of method overloading	... 163
6. Illustration of static members	... 164
7. Illustration of nesting of methods	... 165
8. Illustration of single level inheritance	... 166
9. Illustration of method overriding	... 167
10. Illustration of method overriding	... 168
11. Illustration of passing parameters through constructor	... 170
12. Illustration of method with varargs(variable length arguments)	... 171
13. Return the name	... 172
14. Illustration of public and private access	... 173
<u>15. Passing parameters by values</u>	... 174
<u>16. Passing parameters by reference</u>	... 175
<u>17. Returning an object</u>	... 176
18. Illustration of inner and outer classes	... 177
19. Illustration of abstract	... 178
20. Using abstract methods classes	... 179
21. Illustration of dynamic method dispatch	... 181
5. Array, String and Vectors	... 183—219
<hr/>	
<u>1. Illustration of one dimentional array</u>	... 183
<u>2. Illustration of 1-D variable length array</u>	... 184
3. Reverse a 1-D array	... 185
4. Sorting a list of number in ascending order	... 186
5. Merging two lists of numbers in ascending order both of which are given in ascending order	... 187
6. Insertion in an array at a specific position	... 189
7. Insertion in an array given in ascending order	... 191
8. Deletion of an element from an unsorted array	... 193
9. Deletion of an element from a sorted array	... 196
<u>10. Illustration of 2-D array</u>	... 199
11. Illustration of printing a matrix	... 200

12. Transpose of a matrix	...	201
13. Addition of two matrices	...	202
14. Subtraction of two matrices	...	204
15. Multiplication of two matrices	...	206
16. Illustration of strings	...	210
17. Reversing list of strings	...	211
18. Print some months and their number of days	...	212
19. Sorting of strings in alphabetical order	...	213
20. Illustration of string buffer methods	...	214
21. Manipulation of strings	...	215
22. Illustration of vectors and arrays	...	217
23. Using wrapper class methods	...	218
6. Interfaces (Multiple Inheritance)	...	220—227
1. Illustration of interfaces	...	220
2. Illustration of simple inheritance	...	222
3. Illustration of extended interface	...	223
4. Illustration of multiple classes implements interface	...	224
5. Illustration of printing radius using interface	...	225
6. Illustration of multiple inheritance	...	226
7. Packages (Putting Classes Together)	...	228—230
1. First notepad of package pack	...	228
2. Second notepad of package pack	...	229
3. Third notepad of package pack	...	229
4. Class importing all classes of package pack	...	230
8. Multithreaded Programming	...	231—241
1. Illustration of creation threads using thread class	...	231
2. Illustration of thread method sleep()	...	233
3. Illustration of all thread methods	...	234
4. Illustration of priority of threads using run() method	...	236
5. Illustration of priority of threads	...	237
6. Illustration of use of priority in threads	...	238
7. Illustration of runnable interface	...	240
8. Illustration of runnable interface	...	241

9. Exception Handling	... 242—247
1. Use of try and catch for exception handling	... 242
2. Illustration of catching invalid command line arguments	... 243
3. Illustration of number format exception	... 244
4. Illustration of multiple catch blocks	... 245
5. Illustration of selecting an exception	... 246
6. Illustration of throwing our own exceptions	... 247
10. Applets	... 248—258
1. Illustration of simple applet program	... 248
2. Illustration of passing parameters to an applet	... 250
3. Illustration of displaying numerical values	... 252
4. Illustration of getting input from the user	... 253
5. Drawing nested rectangles using applet	... 255
6. Find greatest of three numbers	... 257
11. Graphics	... 259—281
1. Drawing lines and rectangles	... 259
2. Drawing a square	... 261
3. Illustration of a circle inside the ellipse	... 262
4. Drawing an arc	... 263
5. Illustration of drawing a doll	... 264
6. Drawing a simple polygon with color effects	... 266
7. Drawing polygon using drawPolygon method	... 268
8. Illustration of empty and filled polygons	... 270
9. Illustration of a simple polygon using addPoint method	... 272
10. Applet to draw a line graph	... 274
11. Illustration of loops in applet	... 276
12. Illustration of loops	... 278
13. Illustration of a bar graph	... 280
12. Searching and Sorting	... 282—316
1. Linear search in an unsorted array	... 282
2. Linear search in sorted array(ascending order)	... 285

3. Binary search in an array given in ascending order	...	288
4. Binary search in an array given in descending order	...	291
5. Simple sorting in ascending order	...	294
6. Simple sorting in descending order	...	296
7. Insertion sort for ascending order	...	298
8. Selection sort for ascending order	...	300
9. Bubble sort for ascending order	...	302
10. Quick sort for ascending order	...	305
11. Merge sort for ascending order	...	308
12. Radix sort for ascending order	...	311
13. Shell sort for ascending order	...	314
13. Managing Input Output Files in JAVA	... 317—339	
1. Copying characters from one file to another	...	317
2. Writing bytes to a file	...	319
3. Reading bytes from a file	...	320
4. Copying bytes from one file to another	...	321
5. Reading and writing primitive data	...	323
6. Storing and retrieving integers using data streams	...	324
7. Concatenating and buffering files	...	326
8. Writing and reading with random access	...	328
9. Appending to a text file using random access	...	329
10. Creating files interactively from keyboard input	...	330
11. Creating a file using text fields in windows	...	332
12. Reading a file using text fields	...	336

PREFACE TO THE FIRST EDITION

I am highly delighted to place in the hands of my esteemed readers the book "**Mastering Java Programs**". The motivation came after getting a tremendous success of the books "Mastering C++ Programs" and "Mastering C Programs".

Java is a programming language originally developed by James Gosling at Sun Microsystems and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to bytecode (class file) that can run on any Java Virtual Machine (JVM) regardless of computer architecture.

On 13 November 2006, Sun released much of Java as free and open source software under the terms of the GNU General Public License (GPL). On 8 May 2007 Sun finished the process, making Java's code available under free software/open-source distribution terms, aside from a small portion of code to which Sun did not hold the copyright.

There were five primary goals in the creation of the Java language:

1. It should be "simple, object oriented and familiar".
2. It should be "robust and secure".
3. It should be "architecture neutral and portable".
4. It should execute with "high performance".
5. It should be "interpreted, threaded and dynamic".

Java is an excellent choice for your first programming language. This book is written for people who are computer literate, but have no programming experience. The overall goal of this book is to help users write programs that are :

- Easy to write
- Easy to read
- Easy to understand
- Easy to modify

I have put my sincere efforts and knowledge to make you understand the subject matter in the simplest and easiest form. Valuable suggestions are always most welcome.

WISH YOU A GRAND SUCCESS in your examination, and a very bright future in the field of Computer Science.

—Author



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

ELEMENTARY PROGRAMS

PROGRAM 1

```
/* Illustration of basic arithmetic operators */

class ele1
{
    public static void main(String args[])
    {
        //Arithmetic using integers
        System.out.println("\nInteger arithmetic");
        int i=5+5;
        int j=i*5;
        int k=j/6;
        int l=k-i;
        int m=-l;
        System.out.println("\ni = " +i);
        System.out.println("\nj = " +j);
        System.out.println("\nk = " +k);
        System.out.println("\nl = " +l);
        System.out.println("\nm = " +m);
        //Arithmetic using double
        System.out.println("\nDouble arithmetic");
        double di=5+5;
```

2 MASTERING JAVA PROGRAMS

```
    double dj=di*5;
    double dk=dj/6;
    double dl=dk-i;
    double dm=-dl;
    System.out.println("\ndi = "+di);
    System.out.println("\ndj = "+dj);
    System.out.println("\ndk = "+dk);
    System.out.println("\ndl = "+dl);
    System.out.println("\ndm = "+dm);
}
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele1.java
```

```
C:\jdk1.3\bin>java ele1
```

```
Integer arithmetic
```

```
i = 10
```

```
j = 50
```

```
k = 8
```

```
l = -2
```

```
m = 2
```

```
Double arithmetic
```

```
di = 10.0
```

```
dj = 50.0
```

```
dk = 8.33333333333334
```

```
dl = -1.666666666666666
```

```
dm = 1.666666666666666
```

PROGRAM 2

```
/* Illustration of the bitwise logical operators */

class ele2
{
    public static void main(String args[])
    {
        String binary[]={"1001","1010","1011","1100","1101","1110","1111","0000",
        "0001","0010","0011","0100","0101","0110","0111","1000"};
        int i=5;
        int j=4;
        int k=i;j;
        int l=i&j;
        int m=i^j;
        int n=(~i&j)|(i&~j);
        int o=~i&0x0f;
        System.out.println("\n\ti = "+binary[i]);
        System.out.println("\n\tj = "+binary[j]);
        System.out.println("\n\ti|j = "+binary[k]);
        System.out.println("\n\ti&j = "+binary[l]);
        System.out.println("\n\ti^j = "+binary[m]);
        System.out.println("\n\t~i&j|i&~j = "+binary[n]);
        System.out.println("\n\t~i = "+binary[o]);
    }
}
```

OUTPUT

C:\jdk1.3\bin>javac ele2.java

C:\jdk1.3\bin>java ele2

i = 1110

j = 1101

i;j = 1110

i&j = 1101

i^j = 1010

~i&j|i&~j = 1010

~i = 0011



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PROGRAM 5 —————

```
/* Swap (interchange) two number using a temporary variable */

class ele5
{
    public static void main(String args[])
    {
        int x,y,temp; //variables declared
        x=Integer.parseInt(args[0]); //string converted into integer
        y=Integer.parseInt(args[1]);
        //echo the data
        System.out.println("\nEnterd number are : "+x+" "+y);
        //swapping of numbers
        temp=x;
        x=y;
        y=temp;
        System.out.println("\nSwapped numbers are : "+x+" "+y);
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele5.java

C:\jdk1.3\bin>java ele5 3 43

Entered number are : 3 43

Swapped numbers are : 43 3
```

PROGRAM 6

```
/* Swap (interchange) two numbers without using a temporary variable */

class ele6
{
    public static void main(String args[])
    {
        int x,y; //variables declaration
        x=Integer.parseInt(args[0]);
        y=Integer.parseInt(args[1]);
        //swapping
        y=x+y;
        x=y-x;
        y=y-x;
        System.out.println("\nNumbers after swapping are : "+x+" "+y);
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele6.java
C:\jdk1.3\bin>java ele6 65 87
Numbers after swapping are : 87 65
```

PROGRAM 7

```
/* Find simple interest */
class ele7
{
    public static void main(String args[])
    {
        float principal,rate,time,interest; //Variables declared
        principal=Float.parseFloat(args[0]);
        rate=Float.parseFloat(args[1]); //String converted into float
        time=Float.parseFloat(args[2]);
        //echo the data
        System.out.println("\nPrincipal = "+principal);
        System.out.println("Rate = "+rate);
        System.out.println("Time = "+time);
        interest=principal*rate*time/100;
        System.out.println("\nSimple Interest is : "+interest);
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele7.java

C:\jdk1.3\bin>java ele7 1000 5 4

Principal = 1000.0
Rate = 5.0
Time = 4.0

Simple Interest is : 200.0
```

PROGRAM 8

```
/* Compute surface area and volume of a cube */

class ele8
{
    public static void main(String args[])
    {
        float side,surface_area,volume;// Variables declared
        side=Float.parseFloat(args[0]); // String converted into float
        //echo the data
        System.out.println("\nSide of cube="+side);
        surface_area=6*side*side;
        volume=side*side*side;
        System.out.println("\nSurface area of cube is "+surface_area+
        " sq. units");
        System.out.println("Volume of cube is "+volume+" cubic units");
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele8.java

C:\jdk1.3\bin>java ele8 3

Side of cube=3

Surface area of cube is 54.0 sq.units
Volume of cube is 27.0 cubic units
```



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PROGRAM 13

```
/* Find smallest of three numbers */

class ele13
{
    public static void main(String args[])
    {
        int a,b,c,small;                  //variables declared
        a=Integer.parseInt(args[0]);      //string converted into integer
        b=Integer.parseInt(args[1]);
        c=Integer.parseInt(args[2]);
        //echo the data
        System.out.println("\nGiven numbers are : "+a+" "+b+" "+c);
        small=a;
        if(b<small)
            small=b;
        if(c<small)
            small=c;
        System.out.println("Smallest number is "+small);
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele13.java

C:\jdk1.3\bin>java ele13 26 49 13

Given numbers are : 26 49 13
Smallest number is 13
```

PROGRAM 14 —

```
/* Find biggest of three numbers */

class ele14
{
    public static void main(String args[])
    {
        float num1, num2, num3;          //variables declared
        num1=Float.parseFloat(args[0]);   //string converted into float
        num2=Float.parseFloat(args[1]);
        num3=Float.parseFloat(args[2]);
        //echo the data
        System.out.println("\nGiven numbers are : "+num1+" "+num2+
        "+num3);
        if(num1>num2)
        {
            if(num1>num3)
                System.out.println("\nBiggest number is : "+num1);
            else
                System.out.println("\nBiggest number is : "+num3);
        }
        else
        {
            if(num2>num3)
                System.out.println("\nBiggest number is : "+num2);
            else
                System.out.println("\nBiggest number is : "+num3);
        }
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele14.java

C:\jdk1.3\bin>java ele14 35 87 41

Given numbers are : 35.0 87.0 41.0

Biggest number is : 87.0
```

PROGRAM 15 —————

```

/* Find summation of series  a + ar + ar^2 + ..... + ar^(n-1) */

class ele15
{
    public static void main(String args[])
    {
        float a, r;          //variables declared
        double sum=0.0;
        int n,i;
        a=Float.parseFloat(args[0]);    //string converted into float
        r=Float.parseFloat(args[1]);
        n=Integer.parseInt(args[2]);    //string converted into integer
        //echo the data
        System.out.println("\na: First term : "+a);
        System.out.println("r: Common ratio : "+r);
        System.out.println("n: Number of terms : "+n);
        if(r<1)
            sum=a*(1-Math.pow(r,n))/(1-r);
        else if (r==1)
            sum=n*a;
        else
            sum=a*(Math.pow(r,n)-1)/(r-1);
        System.out.println("\nThe sum of the series is : "+sum);
    }
}

```

OUTPUT

```

C:\jdk1.3\bin>javac ele15.java

C:\jdk1.3\bin>java ele15 1 5 5

a: First term : 1.0
r: Common ratio : 5.0
n: Number of terms : 5

The sum of the series is : 781.0

```

PROGRAM 16 —

```

/* Simulate a simple calculator */

class ele16
{
    public static void main(String args[])
        Throws java.io.IOException
    {
        float a,b,result=0.0f; //variables declared
        char option;
        a=Float.parseFloat(args[0]); //string converted into float
        b=Float.parseFloat(args[1]);
        System.out.println("Enter the operator out of (+,-,*,/)\n");
        option=(char)System.in.read(); //Read a character from keyboard
        switch(option)
        {
            case '+':
                result = a+b;
                System.out.println("\nResult is : "+result);
                break;
            case '-':
                result = a-b;
                System.out.println("\nResult is : "+result);
                break;
            case '*':
                result = a*b;
                System.out.println("\nResult is : "+result);
                break;
            case '/':
                if(b==0)
                {
                    System.out.println("Division by zero is not
possible");
                }
                else
                {
                    result = a/b;
                    System.out.println("\nResult is: "+result);
                }
                break;
            default:
                System.out.println("\nWrong operator entered");
                break;
        }
    }
}

```

18 MASTERING JAVA PROGRAMS

```
    }  
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele16.java  
C:\jdk1.3\bin>java ele16. 30 5  
Enter the operator out of (+, -, *, /)  
+  
Result is : 35.0  
  
C:\jdk1.3\bin>java ele16 30 5  
Enter the operator out of (+, -, *, /)  
-  
Result is : 25.0  
  
C:\jdk1.3\bin>java ele16 30 5  
Enter the operator out of (+, -, *, /)  
*  
Result is : 150.0  
  
C:\jdk1.3\bin>java ele16 30 5  
Enter the operator out of (+, -, *, /)  
/  
Result is : 6.0  
  
C:\jdk1.3\bin>java ele16 30 5  
Enter the operator out of (+, -, *, /)  
?  
Wrong operator entered
```

PROGRAM 17

```
/* Illustration of switch..case control structure
Input a number from 1-7 and write corresponding day of week */

class ele17
{
    public static void main(String args[])
    {
        int day=Integer.parseInt(args[0]); //declaration and instillation
        switch(day)
        {
            case 1 :
                System.out.println("\nWeekday is Sunday");
                break;
            case 2 :
                System.out.println("\nWeekday is Monday");
                break;
            case 3 :
                System.out.println("\nWeekday is Tuesday");
                break;
            case 4 :
                System.out.println("\nWeekday is Wednesday");
                break;
            case 5 :
                System.out.println("\nWeekday is Thursday");
                break;
            case 6 :
                System.out.println("\nWeekday is Friday");
                break;
            case 7 :
                System.out.println("\nWeekday is Saturday");
                break;
            default :
                System.out.println("\nWrong choice");
                break;
        }
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele17.java
C:\jdk1.3\bin>java ele17 3
Weekday is Tuesday
C:\jdk1.3\bin>java ele17 8
Wrong choice
```

PROGRAM 18 —————

```
/* Find area of a triangle and its type */

class ele18
{
    public static void main(String args[])
    {
        float a,b,c,s; //variables declared
        double area;
        a=Float.parseFloat(args[0]); //string converted into float
        b=Float.parseFloat(args[1]);
        c=Float.parseFloat(args[2]);
        //echo the data
        System.out.println("\nThe entered sides are "+a+" "+b+" "+c);
        if(((a+b)>c) && ((a+c)>b) && ((b+c)>a))
        {
            if((a==b) && (a==c))
                System.out.println("\nEquilateral triangle");
            else
                {
                    if((a==b) || (a==c) || (b==c))
                        System.out.println("\nIsosceles triangle");
                    else
                        System.out.println("\nScalene triangle");
                }
            s=(a+b+c)/2.0f;
            area=Math.sqrt(s*(s-a)*(s-b)*(s-c));
            System.out.println("\nArea is "+area+" sq. units");
        }
        else
            System.out.println("\nTriangle not possible");
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele18.java
```

```
C:\jdk1.3\bin>java ele18 3 4 5
```

The entered sides are 3.0 4.0 5.0

Scalene triangle

Area is 6.0 sq. units

```
C:\jdk1.3\bin>java ele18 6 6 6
```

The entered sides are 6.0 6.0 6.0

Equilateral triangle

Area is 15.588457268119896 sq. units

```
C:\jdk1.3\bin>java ele18 4 4 5
```

The entered sides are 4.0 4.0 5.0

Isosceles triangle

Area is 7.806247497997997 sq. units

```
C:\jdk1.3\bin>java ele18 2 1 7
```

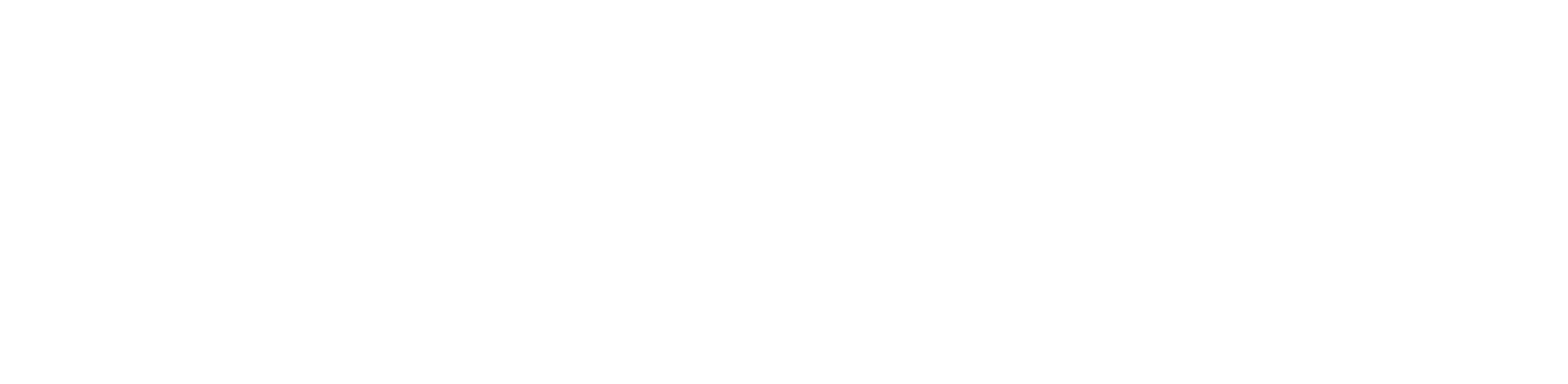
The entered sides are 2.0 1.0 7.0

Triangle not possible

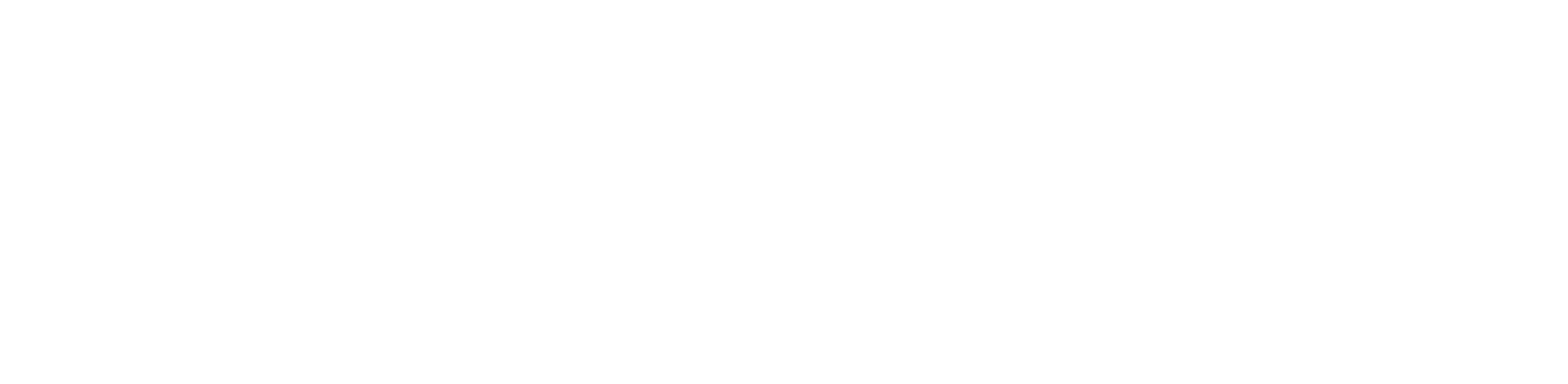
PROGRAM 19

```
/* Find grade of a student */

class ele19
{
    public static void main(String args[])
    {
        int m1,m2,m3,m4,m5,total; //variables declared
        float average;
        m1=Integer.parseInt(args[0]); //string converted into integer
        m2=Integer.parseInt(args[1]);
        m3=Integer.parseInt(args[2]);
        m4=Integer.parseInt(args[3]);
        m5=Integer.parseInt(args[4]);
        //echo the data
        System.out.println("\nEnterd marks are : "+m1+" "+m2+" "+m3+
        "+m4+" "+m5);
        if(m1<35 || m2<35 || m3<35 || m4<35 || m5<35)
            System.out.println("Student can't be given any grade");
        else
        {
            total=m1+m2+m3+m4+m5;
            average=total/5.0f;
            if(average>=80.0f)
                System.out.println("\nStudent has got A grade");
            else if(average>=70.0f)
                System.out.println("\nStudent has got B grade");
            else if(average>=60.0f)
                System.out.println("\nStudent has got C grade");
            else if(average>=50.0f)
                System.out.println("\nStudent has got D grade");
            else if(average>=40.0f)
                System.out.println("\nStudent has got E grade");
            else
                System.out.println("\nStudent has got F grade");
        }
    }
}
```



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

```

    {
        System.out.println("\nImaginary roots\n ");
        r1=r2=-b/(2.0f*a);
        // take square of - disc as square of negative number
        // is not defined on system
        img1=Math.sqrt(-disc)/(2.0f*a);
        img2=-img1;
    }
}

System.out.println("First root is ");
System.out.println("Real part= "+r1+" Img. part= "+img1);
System.out.println("Second root is ");
System.out.println("Real part= "+r2+" Img. part= "+img2);
}
}
}

```

OUTPUT

C:\jdk1.3\bin>javac ele21.java

C:\jdk1.3\bin>java ele21 1 -4 5

Entered co-efficients are : 1.0 -4.0 5.0

Imaginary roots

First root is

Real part= 2.0 Img. part= 1.0

Second root is

Real part= 2.0 Img. part= -1.0

C:\jdk1.3\bin>java ele21 1 5 6

Entered co-efficients are : 1.0 5.0 6.0

Real and distinct roots

First root is

Real part= -2.0 Img. part= 0.0

Second root is

Real part= -3.0 Img. part= 0.0

C:\jdk1.3\bin>java ele21 1 -4 4

Entered co-efficients are : 1.0 -4.0 4.0

Real and equal roots

First root is

Real part= 2.0 Img. part= 0.0

Second root is

Real part= 2.0 Img. part= 0.0

C:\jdk1.3\bin>java ele21 0 2 4

Entered co-efficients are : 0.0 2.0 4.0

Linear equation has single root

Root =-2.0

C:\jdk1.3\bin>java ele21 0 0 5

Entered co-efficients are : 0.0 0.0 5.0

Equation is degenerate

PROGRAM 22 —

```

/* Calculate the area of either circle or rectangle or triangle depending upon the user's choice */

class ele22
{
    public static void main(String args[])
    {
        float area, radius, length, breadth, base, height; //variables declared
        int choice;
        choice=Integer.parseInt(args[0]); //string converted into integer
        switch(choice)
        {
            case 1 :
                System.out.println("\nYou entered the choice for circle");
                radius=Float.parseFloat(args[1]);
                System.out.println("\nThe radius of circle is : "+radius);
                area=3.14f*radius*radius;
                System.out.println("\nArea of circle is : "+area+" sq. units");
                break;
            case 2 :
                System.out.println("\nYou entered the choice for rectangle");
                length=Float.parseFloat(args[1]);
                breadth=Float.parseFloat(args[2]);
                System.out.println("\nThe length and breadth of rectangle are : "+length+"\t"+breadth);
                area=length*breadth;
                System.out.println("\nArea of rectangle is : "+area+" sq. units");
                break;
            case 3 :
                System.out.println("\nYou entered the choice for triangle");
                base=Float.parseFloat(args[1]);
                height=Float.parseFloat(args[2]);
                System.out.println("\nThe base and height of right angled triangle are : "+base+"\t"+height);
                area=(base*height)/2;
                System.out.println("\nArea of right angled triangle is : "+area+" sq. units");
                break;
        }
    }
}

```



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PROGRAM 24

```

/* Calculate the occurrences of positive numbers, negative numbers and zeros in a stream of
data terminated by some specific value */

class ele24
{
    public static void main(String args[])
    {
        int num,i,no_of_pos=0,no_of_neg=0,no_of_zero=0; //variables declared
        System.out.println("\nEnterd stream of data terminated by 9999 is
        : \n");
        i=0;
        while(i<args.length)
        {
            num=Integer.parseInt(args[i]);
            if(num==9999)
                break;
            System.out.print(args[i]+"   ");
            if(num>0)
                no_of_pos++;
            else
            {
                if(num<0)
                    no_of_neg++;
                else
                    no_of_zero++;
            }
            i++;
        }
        System.out.println("\n\nOccurrences of positive numbers : "+no_of_pos);
        System.out.println("\nOccurrences of negetive numbers : "+no_of_neg);
        System.out.println("\nOccurrences of zeros : "+no_of_zero);
    }
}

```

OUTPUT

```

C:\jdk1.3\bin>javac ele24.java
C:\jdk1.3\bin>java ele24 46 38 0 0 -3 -5 -7 0 17 0 -55 9999
Entered stream of data terminated by 9999 is :
46      38      0      0     -3     -5     -7      0      17      0     -55
Occurrences of positive numbers : 3
Occurrences of negetive numbers : 4
Occurrences of zeros : 4

```

PROGRAM 25

/* Print first n natural numbers and their sum using do-while loop */

```
class ele25
{
    public static void main(String args[])
    {
        int n,i=1,sum=0; //variables declared
        n=Integer.parseInt(args[0]); //string converted into integer
        System.out.println("\nFirst "+n+" natural numbers are :\n");
        do
        {
            System.out.print(i+"\t");
            sum+=i;
            i++;
        }
        while(i<=n);
        System.out.println("\nSum = "+sum);
    }
}
```

OUTPUT

C:\jdk1.3\bin>javac ele25.java

C:\jdk1.3\bin>java ele25 10

First 10 natural numbers are :

1 2 3 4 5 6 7 8 9 10

Sum = 55

PROGRAM 26

```
/* Print first n natural numbers and their sum using for loop */

class ele26
{
    public static void main(String args[])
    {
        int n,i,sum=0; //variables declared
        n=Integer.parseInt(args[0]); //string converted into integer
        System.out.println("\nFirst "+n+" natural numbers are :\n");
        for(i=1;i<=n;i++)
        {
            System.out.print(i+"\t");
            sum+=i;
        }
        System.out.println("\n\nSum = "+sum);
    }
}
```

OUTPUT

C:\jdk1.3\bin>javac ele26.java

C:\jdk1.3\bin>java ele26 15

First 15 natural numbers are :

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15					

Sum = 120

PROGRAM 27 —————

```
/* Multiplication table of a given number */

class ele27
{
    public static void main(String args[])
    {
        int num,i; //variables declared
        num=Integer.parseInt(args[0]); //string converted into integer
        System.out.println("\nMultiplication table of "+num+" is :\n");
        for(i=1;i<=10;i++)
        {
            System.out.println(num+" * "+i+" = "+num*i);
        }
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele27.java

C:\jdk1.3\bin>java ele27 15

Multiplication table of 15 is :

15 * 1 = 15
15 * 2 = 30
15 * 3 = 45
15 * 4 = 60
15 * 5 = 75
15 * 6 = 90
15 * 7 = 105
15 * 8 = 120
15 * 9 = 135
15 * 10 = 150
```

PROGRAM 28 —————

```
/* Print first n natural numbers in ascending/descending order */
class ele28
{
    public static void main(String args[])
    {
        int n,num,choice; //variables declared
        n=Integer.parseInt(args[0]); //string converted into integer
        choice=Integer.parseInt(args[1]);
        switch(choice)
        {
            case 1 :
                System.out.println("\n"+n+" Natural numbers in
                                   ascending order are :\n");
                for(num=1;num<=n;num++)
                    System.out.print(num+" ");
                break;
            case 2 :
                System.out.println("\n"+n+" Natural numbers in
                                   descending order are :\n");
                for(num=n;num>=1;num--)
                    System.out.print(num+" ");
                break;
            default :
                System.out.println("\nWrong Choice\n");
                break;
        }
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele28.java
C:\jdk1.3\bin>java ele28 10 1
10 Natural numbers in ascending order are :
1      2      3      4      5      6      7      8      9      10
C:\jdk1.3\bin>java ele28 10 2
10 Natural numbers in descending order are :
10     9      8      7      6      5      4      3      2      1
C:\jdk1.3\bin>java ele28 10 3
Wrong Choice
```

PROGRAM 29

```
/* Count number of 1's in the binary representation of an integer */

class ele29
{
    public static void main(String args[])
    {
        int num,count=0; //variables declared
        num=Integer.parseInt(args[0]); //string converted into integer
        //echo the data
        System.out.println("\nEnterd number is: "+num);
        while(num!=0)
        {
            if(num%2==1)
                count++;
            num/=2;
        }
        System.out.println("\nNumber of 1's in its binary representation : "+count);
    }
}
```

OUTPUT

```
c:\jdk1.3\bin>javac ele29.java

C:\jdk1.3\bin>java ele29 32767

Entered number is: 32767

Number of 1's in its binary representation : 15

C:\jdk1.3\bin>java ele29 1045

Entered number is: 1045

Number of 1's in its binary representation : 4
```

PROGRAM 30

```
/* Convert centigrade temperature to Fahrenheit and vice versa */

class ele30
{
    public static void main(String args[])
    {
        float cent,fahr,value; //variables declared int choice;
        choice=Integer.parseInt(args[0]); //string converted into integer
        value=Float.parseFloat(args[1]); //string converted into float
        switch(choice)
        {
            case1:
                fahr=1.8f*value+32.0f;
                System.out.println("\nCentigrade temperature = "+value);
                System.out.println("\nFahrenheit     temperature = "+fahr);
                break;
            case2:
                cent=(value-32.0f)/1.8f;
                System.out.println("\nFahrenheit     temperature = "+value);
                System.out.println("\nCentigrade     temperature = "+cent);
                break;
            default :
                System.out.println("\nEnter correct choice i.e.1 or 2");
        }
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele30.java
C:\jdk1.3\bin>java ele30 1 37
Centigrade temperature = 37.0
Fahrenheit     temperature = 98.6
C:\jdk1.3\bin>java ele30 2 107
Fahrenheit     temperature = 107.0
Centigrade     temperature = 41.666668
C:\jdk1.3\bin>java ele30 4 98
Enter correct choice i.e.1 or 2
```



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PROGRAM 35

```

/* Find sum of even and odd numbers from 1 to n */

class ele35
{
    public static void main(String args[])
    {
        int n,i,sumeven,sumodd; //variables declared
        n=Integer.parseInt(args[0]); //string converted into integer
        //echo the data
        System.out.println("\nEnterd limit is: "+n);
        sumeven=0;
        sumodd=0;
        for(i=1;i<=n;i++)
        {
            if((i%2)==0)
                sumeven +=i;
            else
                sumodd +=i;
        }
        System.out.println("\nSum of even numbers upto "+n+" is : "
                           "+sumeven);
        System.out.println("\nSum of odd numbers upto "+n+" is : "
                           "+sumodd);
    }
}

```

OUTPUT

```

C:\jdk1.3\bin>javac ele35.java

C:\jdk1.3\bin>java ele35 100

Entered limit is: 100

Sum of even numbers upto 100 is : 2550

Sum of odd numbers upto 100 is : 2500

```

PROGRAM 36

```

/* Read 10 integer from the keyboard and print number of negative and positive integers */

class ele36
{
    public static void main(String args[])
    {
        int i,num,pos_count=0,neg_count=0;           //variables declared
        System.out.println("\nEnterd numbers are : \n");
        for(i=0;i<10;i++)
        {
            num=Integer.parseInt(args[i]);          //string converted into integer
            System.out.print(args[i]+"\t");
            if(num>=0)
                pos_count++;
            else
                neg_count++;
        }
        System.out.println("\nNumber of positive integers is : "+pos_count);
        System.out.println("\nNumber of negative integers is : "+neg_count);
    }
}

```

OUTPUT

C:\jdk1.3\bin>javac ele36.java

C:\jdk1.3\bin>java ele36 16 29 -9 65 -67 -45 21 -5 34 90

Entered numbers are :

16 29 -9 65 -67 -45 21 -5 34 90

Number of positive integers is : 6

Number of negative integers is : 4

PROGRAM 37 —————

```
/* Check an integer for perfect square */

class ele37
{
    public static void main(String args[])
    {
        int num; //variables declared
        double root;
        num=Integer.parseInt(args[0]); //string converted into integer
        root=Math.sqrt(num);
        if(num == root*root)
            System.out.println("\n"+num+" is a perfect square");
        else
            System.out.println("\n"+num+" is not a perfect square");
    }
}
```

OUTPUT

C:\jdk1.3\bin>javac ele37.java

C:\jdk1.3\bin>java ele37 64

64 is a perfect square

C:\jdk1.3\bin>java ele37 40

40 is not a perfect square

PROGRAM 38

```
/* Find all divisor of a positive integer */

class ele38
{
    public static void main(String args[])
    {
        int num,d,mid; //variables declared
        num=Integer.parseInt(args[0]); //string converted into integer
        System.out.println("\nDivisors of " +num+ " are \n");
        mid=num/2;
        for(d=1;d<=mid;d++)
        {
            if(num%d == 0)
                System.out.print(d+"\t");
        }
        System.out.println(num+"\t");
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele38.java

C:\jdk1.3\bin>java ele38 100

Divisors of 100 are

1 2 4 5 10 20 25 50 100

C:\jdk1.3\bin>java ele38 31

Divisors of 31 are

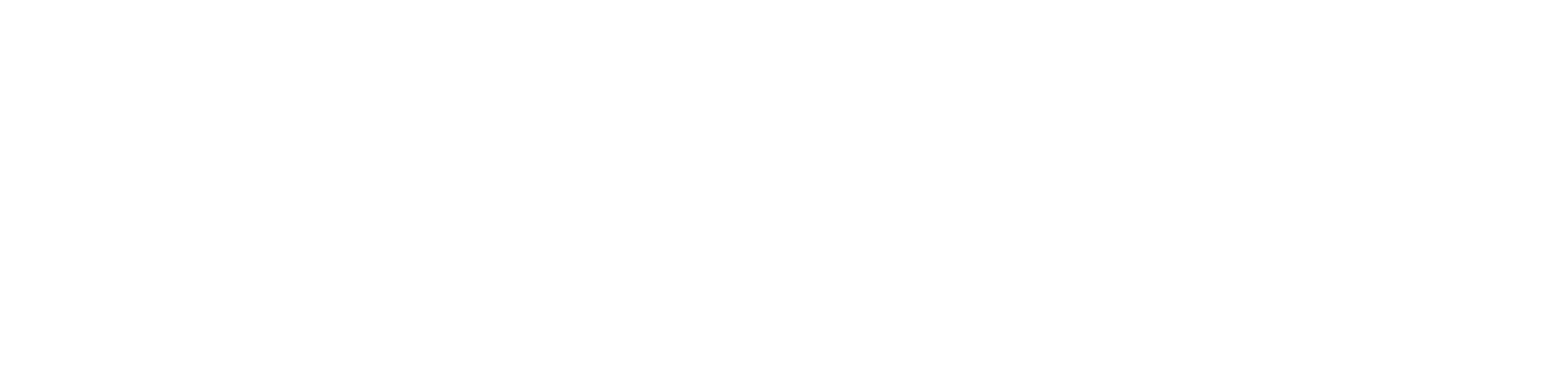
1 31
```



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PROGRAM 43

```
/* Determine all Pythagorean triplets in the range 1 to 100
A Pythagorean triplet is a set of three integer i, j, k,
such that sqrt(i)+sqrt(j)=sqrt(k) */

class ele43
{
    public static void main(String args[])
    {
        int scroll=1,i,j,k; //variables declared
        System.out.println("\nThe Pythagorean triplets in the range 1 to 100
are :\n");
        for(i=1;i<=100;i++)
        {
            for(j=i+1;j<=100;j++)
            {
                for(k=j+1;k<=100;k++)
                {
                    if(Math.pow(i,2)+Math.pow(j,2)== Math.pow(k,2))
                    {
                        System.out.println(i+" , "+j+" , "+k);
                        scroll++;
                        if(scroll%20==0)
                            System.out.println("\nPress any key to continue...");
                    }
                }
            }
        }
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac ele43.java
C:\jdk1.3\bin>java ele43
The Pythagorean triplets in the range 1 to 100 are :
3,4,5
```

5,12,13

6,8,10

7,24,25

8,15,17

9,12,15

9,40,41

10,24,26

11,60,61

12,16,20

12,35,37

13,84,85

14,48,50

15,20,25

15,36,39

16,30,34

16,63,65

18,24,30

18,80,82

Press any key to continue...

20,21,29

20,48,52

21,28,35

21,72,75

24,32,40

24,45,51

24,70,74

25,60,65

27,36,45

28,45,53

28,96,100

30,40,50

30,72,78

32,60,68

33,44,55

33,56,65

35, 84, 91
36, 48, 60
36, 77, 85
39, 52, 65

Press any key to continue...

39, 80, 89
40, 42, 58
40, 75, 85
42, 56, 70
45, 60, 75
48, 55, 73
48, 64, 80
51, 68, 85
54, 72, 90
57, 76, 95
60, 63, 87
60, 80, 100
65, 72, 97



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

```
    }  
}
```

OUTPUT

C:\jdk1.3\bin>javac ele55.java

C:\jdk1.3\bin>java ele55 100

Primes upto 100 are

2	3	5	7	11	13	17	19	23	29
31	37	41	43	47	53	59	61	67	71
73	79	83	89	97					

Count = 25

PROGRAM 56

/* Print the following pattern

```

1
1  2
1  2  3
1  2  3  4
1  2  3  4  5
----- */
class ele56
{
    public static void main(String args[])
    {
        int n,i,j; //variables declared
        n=Integer.parseInt(args[0]); //string converted into integer
        System.out.println("\nDesired pattern with "+n+" lines is \n");
        for(i=1;i<=n;i++)
        {
            for(j=1;j<=i;j++)
                System.out.print(" "+j);
            System.out.print("\n");
        }
    }
}

```

OUTPUT

C:\jdk1.3\bin>javac ele56.java

C:\jdk1.3\bin>java ele56 7

Desired pattern with 7 lines is

```

1
1  2
1  2  3
1  2  3  4
1  2  3  4  5
1  2  3  4  5  6
1  2  3  4  5  6  7

```

PROGRAM 57 —————

/* Print the following pattern

```
1
2  2
3  3  3
4  4  4  4
-----
class ele57
{
    public static void main(String args[])
    {
        int n,i,j; //variables declared
        n=Integer.parseInt(args[0]); //string converted into integer
        System.out.println("\nDesired pattern with "+n+" lines is \n");
        for(i=1;i<=n;i++)
        {
            for(j=1;j<=i;j++)
                System.out.print(" "+i);
            System.out.print("\n");
        }
    }
}
```

OUTPUT

C:\jdk1.3\bin>javac ele57.java

C:\jdk1.3\bin>java ele57 6

Desired pattern with 6 lines is

```
1
2  2
3  3  3
4  4  4  4
5  5  5  5  5
6  6  6  6  6  6
```



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Chapter 2

METHODS

PROGRAM 1

```
/* Display general message using method */

class fun
{
    /* Method definition display_message() */
    void display_message()
    {
        System.out.println("\nWelcome to the world of JAVA programming\n");

        System.out.println("Programming is nothing but logic implementation");
    }
}

class met1
{
    public static void main(String args[])
    {
        fun obj1=new fun();          //Creating object
        obj1.display_message();     //method called
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac met1.java
C:\jdk1.3\bin>java met1
Welcome to the world of JAVA programming
Programming is nothing but logic implementation
```

PROGRAM 2

```
/* Calculate compound interest using method */

class met2
{
    public static void main(String args[])
    {
        float principal,rate,time;      //variables declared
        principal=Float.parseFloat(args[0]); //string converted into float
        rate=Float.parseFloat(args[1]);
        time=Float.parseFloat(args[2]);
        // echo the data
        System.out.println("\nPrincipal = "+principal+"\nRate =
        "+rate+"\nTime = "+time);
        fun obj1=new fun();           //object created
        obj1.calculate(principal,rate,time); //method called
    }
}

class fun
{
    //method definition calculate()
    void calculate(float p,float r,float t)
    {
        double interest;
        interest=p*(Math.pow((1+r/100),t))-p;
        System.out.println("\nCompound interest is : "+(float)interest);
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac met2.java

C:\jdk1.3\bin>java met2 1000 5 2

Principal = 1000.0
Rate = 5.0
Time = 2.0

Compound interest is : 102.4999
```

PROGRAM 3

```
/* Add three numbers using method */

class met3
{
    public static void main(String args[])
    {
        float a,b,c,sum;           //variables declared
        a=Float.parseFloat(args[0]); //string converted into integer
        b=Float.parseFloat(args[1]);
        c=Float.parseFloat(args[2]);
        //echo the date
        System.out.println("\nEnterd numbers are : "+a+"\t"+b+"\t"+c);
        fun obj1=new fun();         //object created
        sum=obj1.add(a,b,c);       //method called
        System.out.println("\nSum of three numbers is : "+sum);
    }
}

class fun
{
    /* Method definition add() */
    float add(float a,float b,float c)
    {
        return(a+b+c);
    }
}
```

OUTPUT

C:\jdk1.3\bin>javac met3.java

C:\jdk1.3\bin>java met3 10 35 25

Entered numbers are : 10.0 35.0 25.0

Sum of three numbers is : 70.0



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PROGRAM 19 —————

```
/* Sum of the series x-(x^2)/2!+(x^3)/3!-(x^4)/4!+... */

class met19
{
    public static void main(String args[])
    {
        float x,term,sum=0.0f;      //variables declared
        int n=16,sign=-1,i;
        x=Float.parseFloat(args[0]); //string converted into float
        //echo the data
        System.out.println("\nEnterd value of x is : "+x);
        sum=x;
        fun obj1=new fun();          //object created
        for(i=2;i<=n;i++)
        {
            term=(float)Math.pow(x,i)/obj1.factorial(i); //method called
            term*=sign;
            sum+=term;
            sign*=-1; //keep rotating the sign
        }
        System.out.println("\nSum of the series is : "+sum);
    }
}

class fun
{
    /*method definition factorial() */
    long factorial(int n)
    {
        long prod=1;
        int i;
        for(i=1;i<=n;i++)
            prod*=i;
        return(prod);
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac met19.java
C:\jdk1.3\bin>java met19 2
Entred value of x is : 2.0
Sum of the series is : 0.86466485
```

PROGRAM 20

```
/* Compute sum of the series x+x^2/3!+x^3/5!+...+x^n(2n-1) */
class met20
{
    public static void main(String args[])
    {
        float x=Float.parseFloat(args[0]);      //string converted into float
        int n=Integer.parseInt(args[1]);        //string converted into integer
        //echo the data
        System.out.println("\nEnterd value of x and n are : "+x+"\t"+n);
        fun obj1=new fun();                  //object created
        System.out.println("\nSum of the series is : "+obj1.sumfun(x,n));
        // method called
    }
}

class fun
{
    /* method definition sum() */
    float sumfun(float x,int n)
    {
        float sum=x;
        int i;
        //method fact called(nesting of methods)
        for(i=2;i<=n;i++)
            sum+=Math.pow(x,i)/fact(2*i-1);
        return(sum);
    }
    /* method definition fact() */
    long fact(int n)
    {
        long prod=1;
        int i;
        for(i=1;i<=n;i++)
            prod*=i;
        return(prod);
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac met20.java
C:\jdk1.3\bin>java met20 1 6
Entered value of x and n are : 1.0      6
Sum of the series is : 1.175201
```

PROGRAM 21 —

```

/* Compute sum of series y+y^3/2!+y^5/3!+...+y^(2m-1)/m! */

class met21
{
    public static void main(String args[])
    {
        float y=Float.parseFloat(args[0]); //string converted into float
        int m=Integer.parseInt(args[1]); //string converted into integer
        //echo the data
        System.out.println("\nEnterd value of y and m are : "+y+"\t"+m);
        fun obj1=new fun();           //object created
        System.out.println("\nSum of the series is : "+obj1.sumfun(y,m));
        // method called
    }
}
class fun
{
    /* method definition sumfun() */
    float sumfun(float y,int m)
    {
        float sum=y;
        int i;
        //method fact called(nesting of methods)
        for(i=2;i<=m;i++)
            sum+=Math.pow(y,2*i-1)/fact(i);
        return(sum);
    }
    /* method definition fact() */
    long fact(int n)
    {
        long prod=1;
        int i;
        for(i=1;i<=n;i++)
            prod*=i;
        return(prod);
    }
}

```

OUTPUT

```

C:\jdk1.3\bin>javac met21.java
C:\jdk1.3\bin>java met21 1 6
Entered value of y and m are : 1.0      6
Sum of the series is : 1.7180555

```



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

```
var=tempsum/n;
// find the standard deviation
sd=(float)Math.sqrt(var);           //type casting
return(sd);
}
}
```

OUTPUT

```
C:\jdk1.3\bin>javac met26.java
C:\jdk1.3\bin>java met26 5 12 8 15 13 22
5 entered numbers are :
12.0      8.0      15.0      13.0      22.0
Standard deviation = 4.604346
```

PROGRAM 27 —

```

/* Display word equivalent of a number (of 1-3 digits) */

class met27
{
    public static void main(String args[])
    {
        int num,temp=0,digit=0; //variables declared
        num=Integer.parseInt(args[0]); //string converted into integer
        temp=num;
        while(temp>=1)
        {
            temp /=10;
            digit++;
            if(digit>3)
            {
                System.out.println("\nYou have entered more than three digit
integer\n");
            }
        }
        fun obj1=new fun(); //object created
        System.out.println("\nYou have entered : \n");
        switch(digit)
        {
            case 1 : obj1.onedigit(num); //method called
            break;
            case 2 : obj1.twodigit(num); //method called
            break;
            case 3 : obj1.threedigit(num); //method called
            break;
            default : System.out.print("Zero");
            break;
        }
    }
}

class fun
{
    //method definition threedigit()
    void threedigit(int n)
    {
        int quotient,rem;
        quotient=n/100;
        rem=n%100;
    }
}

```

```
onedigit(quotient);      //method called
if(quotient==0)
    twodigit(rem);      //method called
else
{
    System.out.print("HUNDRED   ");
    twodigit(rem);      //method called
}
}

//method definition twodigit()
void twodigit(int n)
{
    int quotient,rem;
    quotient=n/10;
    rem=n%10;
    if(quotient==1)
    {
        twosp(n);          //method called
        return;
    }
    twoty(quotient);      //method called
    if(rem==0)
        System.out.print("   ");
    else
        onedigit(rem);      //method called
}
// method definition onedigit()
void onedigit(int n)
{
    switch(n)
    {
        case 1 : System.out.print("ONE   ");
                   break;
        case 2 : System.out.print("TWO   ");
                   break;
        case 3 : System.out.print("THREE   ");
                   break;
        case 4 : System.out.print("FOUR   ");
                   break;
        case 5 : System.out.print("FIVE   ");
                   break;
        case 6 : System.out.print("SIX   ");
                   break;
        case 7 : System.out.print("SEVEN   ");
                   break;
    }
}
```

```
case 8 : System.out.print("EIGHT    ");
           break;
case 9 : System.out.print("NINE    ");
           break;
}
}

//method definition twosp()
void twosp(int n)
{
    switch(n)
    {
        case 11 : System.out.print("ELEVEN    ");
                   break;
        case 12 : System.out.print("TWELVE    ");
                   break;
        case 13 : System.out.print("THIRTEEN   ");
                   break;
        case 14 : System.out.print("FOURTEEN   ");
                   break;
        case 15 : System.out.print("FIFTEEN   ");
                   break;
        case 16 : System.out.print("SIXTEEN   ");
                   break;
        case 17 : System.out.print("SEVENTEEN  ");
                   break;
        case 18 : System.out.print("EIGHTEEN  ");
                   break;
        case 19 : System.out.print("NINETEEN  ");
                   break;
    }
}

//method definition twoty()
void twoty(int n)
{
    switch(n)
    {
        case 2 : System.out.print("TWENTY    ");
                   break;
        case 3 : System.out.print("THIRTY    ");
                   break;
        case 4 : System.out.print("FOURTY   ");
                   break;
        case 5 : System.out.print("FIFTY    ");
                   break;
    }
}
```

```
    case 6 : System.out.print("SIXTY   ");
               break;
    case 7 : System.out.print("SEVENTY   ");
               break;
    case 8 : System.out.print("EIGHTY   ");
               break;
    case 9 : System.out.print("NINETY   ");
               break;
}
}
}
```

OUTPUT

C:\jdk1.3\bin>javac met27.java

C:\jdk1.3\bin>java met27 397

You have entered :

THREE HUNDRED NINETY SEVEN

C:\jdk1.3\bin>java met27 12

You have entered :

TWELVE

C:\jdk1.3\bin>java met27 0

You have entered :

Zero

PROGRAM 28

```
/* Read two integer numbers and find their sum, difference, multiplication and division using
a separate method for each of these operations */

class met28
{
    public static void main(String args[])
    {
        int num1,num2; //variables declared
        num1=Integer.parseInt(args[0]); //string converted into integer
        num2=Integer.parseInt(args[1]);
        //echo the data
        System.out.println("\nEnterd numbers are : "+num1+"\t"+num2);
        fun obj1=new fun();           //object created
        obj1.add(num1,num2);         //method called
        obj1.sub(num1,num2);         //method called
        obj1.multiply(num1,num2);    //method called
        obj1.divide(num1,num2);      //method called
    }
}

class fun
{
    //method definition add()

    void add(int value1,int value2)
    {
        int result=value1+value2;
        System.out.println("\nSum = "+result);
    }

    //method definition sub()

    void sub(int value1,int value2)
    {
        int result=value1-value2;
        System.out.println("\nDifference = "+result);
    }

    //method definition multiply()

    void multiply(int value1,int value2)
    {
        int result=value1*value2;
        System.out.println("\nMultiplication = "+result);
    }
}
```

```
}

//method definition divide()

void divide(int value1,int value2)
{
    int result=value1/value2;
    if(value2==0)
        System.out.println("\nDivision by 0 not possible");
    else
        System.out.println("\nAfter integer division, quotient = "+result);
}
}
```

OUTPUT

```
C:\jdk1.3\bin>javac met28.java

C:\jdk1.3\bin>java met28 500 25

Entered numbers are : 500      25

Sum = 525

Difference = 475

Multiplication = 12500

After integer division, quotient = 20
```



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PROGRAM 5

```

/* Gcd of two positive integers (highest common factor) */

class rec5
{
    public static void main(String args[])
    {
        int m,n,result; //variables declared
        m=Integer.parseInt(args[0]); //string converted into integer
        n=Integer.parseInt(args[1]);
        fun obj1=new fun(); //object created
        result=obj1.gcd(m,n); //method called
        System.out.println("\nGreatest common divisor of "+m+" and "+n+
                           " = "+result);
    }
}

class fun
{
    /* method definition gcd() */
    int gcd(int m,int n)
    {
        if(m%n==0)
            return(n);
        else
            return(gcd(n,m%n));
    }
}

```

OUTPUT

C:\jdk1.3\bin>javac rec5.java

C:\jdk1.3\bin>java rec5 12 46

Greatest common divisor of 12 and 46 = 2

C:\jdk1.3\bin>java rec5 31 20

Greatest common divisor of 31 and 20 = 1

PROGRAM 6

```

/* Generate first n Fibonacci terms */

class rec6
{
    public static void main(String args[])
    {
        int i,n;      //variables declared
        n=Integer.parseInt(args[0]); //string converted into integer
        System.out.println("\n"+n+" fibonacci terms are\n");
        fun obj1=new fun(); //object created
        for(i=1;i<=n;i++)
            System.out.print(obj1.fib(i)+"\t"); //method called
    }
}

class fun
{
    /* method definition fib() */
    long fib(int n)
    {
        if(n==1)
            return(0);
        else
        {
            if(n==2)
                return(1);
            else
                return(fib(n-1)+fib(n-2));
        }
    }
}

```

OUTPUT

C:\jdk1.3\bin>javac rec6.java

C:\jdk1.3\bin>java rec6 30

30 fibonacci terms are

0	1	1	2	3	5	8	13	21	34
55	89	144	233	377	610	987	1597	2584	4181
6765	10946	17711	28657	46368	75025	121393	196418	317811	514229

PROGRAM 7

```
/* Add two positive integers using recursion */

class rec7
{
    public static void main(String args[])
    {
        int x,y; //variables declared
        x=Integer.parseInt(args[0]); //string converted into integer
        y=Integer.parseInt(args[1]);
        //echo the data
        System.out.println("\nEnterd integers are : "+x+"\t"+y);
        fun obj1=new fun(); //object created
        System.out.println("\nSum = "+obj1.add(x,y)); //method called
    }
}

class fun
{
    /* method definition add() */
    int add(int x,int y)
    {
        if(x==0)
            return(y);
        else
            return(add(--x,++y));
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac rec7.java

C:\jdk1.3\bin>java rec7 3 7

Entered integers are : 3 7

Sum = 10
```

PROGRAM 8

```
/* Multiply two positive integers using recursion */

class rec8
{
    public static void main(String args[])
    {
        int a,b; //variables declared
        a=Integer.parseInt(args[0]); //string converted into integer
        b=Integer.parseInt(args[1]);
        //echo the data
        System.out.println("\nEntered two integers are : "+a+"\t"+b);
        fun obj1=new fun(); //object created
        System.out.println("\nProduct = "+obj1.prod(a,b)); //method called
    }
}

class fun
{
    /* method definition prod() */
    long prod(int a,int b)
    {
        if((a==0 || b==0))
            return(0);
        if(b==1)
            return(a);
        else
            return(prod(a,b-1)+a);
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac rec8.java

C:\jdk1.3\bin>java rec8 50 6

Entered two integers are : 50      6

Product = 300
```



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PROGRAM 11

```

/* Compute the positive integer or negative integers exponential power of floating point number*/

class rec11
{
    public static void main(String args[])
    {
        float x; //variables declared
        int n;
        x=Float.parseFloat(args[0]); //string converted into float
        n=Integer.parseInt(args[1]); //string converted into integer
        // echo the data
        System.out.println("\nThe mantissa and exponent part are :
        "+x+"\t"+n);
        fun obj1=new fun(); //object created
        if(n<0)
            System.out.println("\nResult = "+1/obj1.pow_fun(x,n) );//method called
        else
            System.out.println("\nResult = "+obj1.pow_fun(x,n) );//method called
    }
}

class fun
{
    /* method definition pow_fun() */
    float pow_fun(float x,int n)
    {
        if(n==0)
            return(1);
        else
        {
            if(n<0)
                return(x*pow_fun(x,n+1) );
            else
                return(x*pow_fun(x,n-1) );
        }
    }
}

```

OUTPUT

```
C:\jdk1.3\bin>javac rec11.java
```

```
C:\jdk1.3\bin>java rec11 2.5 4
```

```
The mantissa and exponent part are : 2.5      4
```

```
Result = 39.0625
```

```
C:\jdk1.3\bin>java rec11 2.0 -4
```

```
The mantissa and exponent part are : 2.0      -4
```

```
Result = 0.0625
```

PROGRAM 12

```

/* Evaluate f(x) = 0           x = 1
   f(x/2)+1    if   x > 1  */

class rec12
{
    public static void main(String args[])
    {
        int x; //variables declared
        x=Integer.parseInt(args[0]); //string converted into integer
        //echo the data
        System.out.println("\nValue of x = "+x);
        fun obj1=new fun();          //object created
        System.out.println("\nf(\""+x+"\") = "+obj1.f(x)); //method call
    }
}

class fun
{
    /* method definition f() */
    int f(int x)
    {
        if(x==1)
            return(0);
        else
            return(f(x/2)+1);
    }
}

```

OUTPUT

C:\jdk1.3\bin>javac rec12.java

C:\jdk1.3\bin>java rec12 20

Value of x = 20

f(20) = 4

C:\jdk1.3\bin>java rec12 150

Value of x = 150

f(150) = 7



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PROGRAM 15

/* Passing parameters by values */

```
class call_value
{
    public static void main(String args[])
    {
        check obj1=new check();
        int x=45;
        int y=34;
        System.out.println("\nx and y before call : "+x+" and "+y);
        obj1.method(x,y);
        System.out.println("\nx and y after call : "+x+" and "+y);
    }
}
class check
{
    void method(int x,int y)
    {
        x*=5;
        y/=5;
    }
}
```

OUTPUT

C:\jdk1.3\bin>javac call_value.java

C:\jdk1.3\bin>java call_value

x and y before call : 45 and 34

x and y after call : 45 and 34

PROGRAM 16 —

```
/* Passing parameters by reference */

class call_reference
{
    public static void main(String args[])
    {
        check obj1=new check(34,45);
        System.out.println("\nx and y before call : "+obj1.x+" and "
        "+obj1.y);
        obj1.method(obj1);
        System.out.println("\nx and y after call : "+obj1.x+" and "+obj1.y);
    }
}

class check
{
    int x,y;
    check(int i,int j)
    {
        x=i;
        y=j;
    }
    // pass an object
    void method(check obj)
    {
        obj.x *=5;
        obj.y /=5;
    }
}
```

OUTPUT

```
C:\jdk1.3\bin>javac call_reference.java

C:\jdk1.3\bin>java call_reference

x and y before call : 34 and 45

x and y after call : 170 and 9
```



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

ABOUT THE BOOK

The book **Mastering Java Programs** has been written for the students of Engineering, M.Tech., MCA, M.Sc., MIT, BCA, C-DAC, DOEACC-A Level, BIT, B.Sc., PGDCA and other computer courses. This book shows you everything you require to run Java Programs. Logical arrangements of contents, clarity of presentation aid the student and novice in mastering the Java language. A number of unique features make this book different from other existing books in the field of Java programming:

- Motivates the unmotivated.
- Over 220 tested program examples provided.
- User friendly approach to enhance the user's understanding of practical concepts.
- Complete coverage of the Java language's fundamental programming principles using Java programs.
- All the programs have been thoroughly tested.
- Essential for every Java learner and programmer.

ABOUT THE AUTHOR

J. B. Dixit is a high profile author. He has an avid interest in seeing students become well educated—especially in Information Technology, Programming and Mathematics. He spends his time watching what's happening in business and society and on college/university campuses and listening to the views expressed by lecturers, students and other participants in the Computer and Mathematics revolution. He then tries to translate those observations into meaningful language that can be best understood by students.

Over the past decade, he has **authored more than 40 quality books**, most of them on Computers, Information Technology and Mathematics for Engineering level courses.

He has been teaching Engineering, M.Tech., M.C.A., M.B.A., M.Sc., B.C.A., B.I.T., B.B.A. and other diploma course students for the last seventeen years. Many of his students are very well placed in IT industry all over the world (including GLU and Microsoft Corporation, USA). A logically sound and experienced teacher, recipient of various honours and scholarships, he has few more books in the pipeline for various Engineering and Degree level courses.



**FIREWAL
MEDIA**

ISBN 978-93-80298-38-2



9 789380 298382

Copyrighted material