**Chapter 8**

**Recursion Practice**

**(and still more)**

11. For each call to the following method, indicate what value is returned.

public int mystery1(int x, int y)

{

if (x < y)

return x;

else

return mystery1(x-y, y);

}

mystery1(6, 13)

|  |
| --- |
|  |

mystery1(8, 2)

|  |
| --- |
|  |

mystery1(14, 10)

|  |
| --- |
|  |

12. For each call to the following method, indicate what console output is produced.

public void mystery2(int n)

{

if (n <= 1)

System.out.print(n);

else

{

mystery2(n/2);

System.out.print(“,” + n);

}

}

mystery2(1)

|  |
| --- |
|  |

mystery2(4)

|  |
| --- |
|  |

mystery2(100)

|  |
| --- |
|  |

13. For each call to the following method, indicate what value is returned.

public int mystery3(int n)

{

if (n < 0)

return –mystery3(-n);

else if (n < 10)

return n;

else

return mystery3(n/10 + n%10);

}

mystery3(6)

|  |
| --- |
|  |

mystery3(17)

|  |
| --- |
|  |

mystery3(-479)

|  |
| --- |
|  |

14. For each call to the following method, indicate what value is returned.

public int mystery4(int n)

{

if(n < 0)

return mystery4(-n);

else if (n < 10)

return n;

else

return n%10 + mystery4(n/10);

}

mystery4(8)

|  |
| --- |
|  |

mystery4(-52)

|  |
| --- |
|  |

mystery4(3052)

|  |
| --- |
|  |

15. Assume the array contains: {2,4,6} and that the call to the sum method is: sum(arr,3). What value is returned?

int sum(int arr[], int n)

{

if (n == 0)

return 0;

else

{

int smallResult = sum(arr, n-1);

return smallResult + arr[n-1];

}

}

16. For each call to the following method, indicate what value is returned.

public int mystery5(int x, int y)

{

if (x < 0)

return –mystery5(-x, y);

else if (y < 0)

return –mystery5(x, -y);

else if (x ==0 && y == 0)

return 0;

else

return 100\*mystery5(x/10, y/10) + 10\*(x%10) + y%10;

}

mystery5(5,7)

|  |
| --- |
|  |

mystery5(12,9)

|  |
| --- |
|  |

mystery5(-7,4)

|  |
| --- |
|  |

17. For each call to the following method, indicate what value is returned.

public void mystery6(int x, int y)

{

if(y == 1)

System.out.print(x);

else

{

System.out.print(x\*y + “,”);

mystery6(x, y-1);

System.out.print(“,” + x\*y);

}

}

mystery6(4,1)

|  |
| --- |
|  |

mystery6(8,2)

|  |
| --- |
|  |

mystery6(3,4)

|  |
| --- |
|  |

18. For each call to the following method, indicate what console output is produced.

public void mystery7(int n)

{

if(n <= 0)

System.out.print(“\*”);

else if (n%2 == 0)

{

System.out.print(“(“);

mystery7(n-1);

System.out.print(“)”);

}

else

{

System.out.print(“[“);

mystery7(n-1);

System.out.print(“]”);

}

}

mystery7(0)

|  |
| --- |
|  |

mystery7(1)

|  |
| --- |
|  |

mystery7(5)

|  |
| --- |
|  |

19. For each call to the following method, indicate what console output is produced.

public void mystery8(int n)

{

if(n > 100)

System.out.print(n);

else

{

mystery8(2\*n);

System.out.print(“,” + n);

}

}

mystery8(113)

|  |
| --- |
|  |

mystery8(70)

|  |
| --- |
|  |

mystery8(42)

|  |
| --- |
|  |

20.For each call to the following method, indicate what console output is produced.

public void mystery9(int x)

{

if(x < 10)

System.out.print(x);

else

{

int y = x%10;

System.out.print(y);

mystery9(x/10);

System.out.print(y);

}

}

mystery9(7)

|  |
| --- |
|  |

mystery9(38)

|  |
| --- |
|  |

mystery9(194)

|  |
| --- |
|  |