

Recursion Day 1

APCS MC Practice

```
public int f1 (int a)
{
    if (a <= 1){ return a; }
    else { return a + f1(a - 2); }
}
```

f1(1) =

f1(4) =

f1(8) =

```
public int f2 (int a)
{
    if (a == 0) { return; }
    else {
        System.out.println(a);
        f2(a-1);
    }
}
```

f2(5) outputs:

```
public int f3 (int a)
{
    if (a == 0) { return; }
    else {
        f3(a-1);
        System.out.println(a);
    }
}
```

f3(5) outputs:

8. What value is returned by the call something(4, 6)? _____

- (A) 4
- (B) 6
- (C) 24
- (D) 1296
- (E) 4096

```
public int something (int a, int b)
{
    if (b <= 1)
    {
        return a;
    }
    else
    {
        return something (a, b-1);
    }
}
```

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) _____

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) _____

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) _____

30. Consider the following recursive method: _____

```
public static void printStars (int k)
{
    if (k>0)
    {
        printStars(k-1);
        for (int j=1; j<=k; j++)
            System.out.print("*");
        System.out.println();
    }
}
```

What is the output as a result of the call `printStars(4)`?

- | | | | |
|-----|------------------------|-----|------------------|
| (A) | ****

**
* | (D) | *
**
*** |
| (B) | *
**

**** | (E) | *
*
*
* |
| (C) | ***
**
* | | |

31. Consider the following recursive method:

```
public int mystery (int k)
{
    if (k == 1)
        return 0;
    else
        return (1 + mystery (k/2));
}
```

What value is returned by the call `mystery(16)`?

- (A) 0
- (B) 2
- (C) 4
- (D) 5
- (E) 16

33. Questions 33 and 34 refer to the following recursive method:

```
public static int compute (int x, int y)
{
    if (x == y)
        return x;
    else
        return (compute(x+1, y-1));
}
```

What is returned by the call `compute(1, 5)`?

- (A) 1
- (B) 2
- (C) 3
- (D) 4
- (E) No value is returned because infinite recursion occurs.

34. Which of the following calls leads to an infinite recursion?

- I. `compute(2, 8)`
- II. `compute(8, 2)`
- III. `compute(2, 5)`

- (A) I only
- (B) II only
- (C) III only
- (D) I and II
- (E) II and III

36. Consider the following method:

```
public void mystery (int a, int b)
{
    System.out.print (a + " ");
    if (a <= b)
        mystery (a + 5, b -1);
}
```

What is the output when `mystery (0, 16)` is called?

- (A) 0
- (B) 0 5
- (C) 0 5 10
- (D) 0 5 10 15
- (E) 0 5 10 15 20

37. What is the output when `smile (4)` is called?

```
public static void smile (int n)
{
    if (n==0)
        return;
    for (int k=1; k<=n; k++)
        System.out.print ("smile!");
    smile(n-1);
}
```

- (A) smile!
- (B) smile!smile!
- (C) smile!smile!smile!
- (D) smile!smile!smile!smile!
- (E) smile!smile!smile!smile!smile!smile!smile!smile!smile!smile!smile!

42. Consider the following method:

```
public int getSomething(int value)
{
    if(value < 1)
        return 0;
    else
        return 1 + getSomething(value-1) + getSomething(value-2);
}
```

What is returned by the call `getSomething (4)`?

- (A) 0
- (B) 1
- (C) 2
- (D) 5
- (E) 7

44. What is the output by the call `fun (3)`?

```
public void fun (int x)
{
    if (x>=1)
    {
        System.out.print(x);
        fun (x-1);
    }
}
```

- (A) 3 2 1
- (B) 1 2 3
- (C) 2 3
- (D) 3 2 1 0
- (E) Nothing will be printed due to an infinite recursion

49. What is the output by the call `fun (3)`?

```
public void fun (int x)
{
    if (x<1)
    {
        System.out.print(x);
    }
    else
    {
        System.out.print(x);
        fun (x-1);
    }
}
```

- (A) 3 2 1 0 3 2 1 0
- (B) 3 2 1 0
- (C) 3 2 1 0 0 1 2 3
- (D) 0 1 2 3
- (E) Nothing will be printed due to infinite recursion

56. Consider the following method:

```
//precondition: num>=0

public static void mystery (int num)
{
    if (num >1)
        mystery (num/2);
    System.out.print (num%2);
}
```

What is the best postcondition for `mystery`?

- (A) Reverses the digits of `num`
- (B) Prints the remainder when `num` is divided by 2
- (C) Prints one-half `num`
- (D) Prints the square root of `num`.
- (E) Prints the binary representation of `num`.