

Computer Science 245
Homework 1
Algorithm Analysis I
Due Wednesday, February 1st, 2017

Give $\Theta()$ running times for each of the following functions, as a function of their input parameter. HINT – if you are having trouble with finding $\Theta()$, first find $O()$, then $\Omega()$. For partial credit, just find $O()$.

1. (2 points)

```
public static int f1(int n)
{
    int sum = 0;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++)
            sum++;
        for (int j = 0; j < n; j++)
            sum++;
        for (int j = 0; j < n; j++)
            sum++;
    }
    return sum;
}
```

2. (2 points)

```
public static int f2(int n)
{
    int sum = 0;
    sum += f1(n) * f1(n);
    sum += f1(n/2);
    return sum;
}
```

3. (2 points)

```
public static int f3(int n)
{
    int sum = 0;
    int tmp = 0;
    while (tmp < n * n * n)
    {
        for (int i = n; i > 0; i--)
        {
            sum++;
        }
        for (int i = n; i > 1; i = i / 2)
        {
            sum++;
        }
        tmp+= 2;
    }
    return sum;
}
```

4. (2 points)

```
public static int f4(int n)
{
    int sum = 0;
    for (int i = 0; i < n * n; i++)
    {
        for (int j = 0; j < n * n; j++)
        {
            sum++;
        }
        for (int j = 0; j < n; j++)
        {
            sum++;
        }
    }
    return sum;
}
```

5. (2 points)

```
public static int f5(int n)
{
    int sum = 0;

    for (int i = 0; i < n; i++)
        sum++;

    while(n > 1)
    {
        sum++;
        n = n / 4;
    }

    return sum;
}
```

6. (2 points)

```
public static int f6(int n)
{
    int sum = 0;
    for (int i = 1; i < n * n; i *= 2)
    {
        sum++;
    }
    return sum;
}
```

7. (2 points)

```
public static int f7(int n)
{
    int sum = 0;
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j += 2)
            sum++;
        for (int j = n; j > 1; j /= 2)
            sum++;
        for (int j = 0; j < j; j += 2)
            sum++;
        for (int j = 1; j < n; j *= 2)
            sum++;
    }
    return sum;
}
```

8. (2 points)

```
public static int f8(int n)
{
    int sum = 0;

    for (int i = 0; i < n; i++)
        for (int j = n; j > 0; j -= 2)
            for (int k = n; k > 1; k /= 2)
                for (int l = 1; l < n; l *= 2)
                    sum++;

    return sum;
}
```

9. (2 points)

```
public static int f9(int n)
{
    int sum = 0;

    int tmp1 = n;
    while (tmp1 > 1) {
        int tmp2 = 0;
        while (tmp2 < n/2) {
            tmp2++;
            sum++;
        }
        tmp2 = n/2;
        while (tmp2 > 0) {
            tmp2--;
            sum++;
        }
        tmp1 = tmp1/2;
    }

    return sum;
}
```

10. (2 points)

```
public static int f10(int n)
{
    return f1(n*n*n)
}
```

11. (2 points)

```
public static int f11(int n)
{
    int sum = 0;
    sum += f1(n);
    sum += f2(n);
    sum += f3(n);
    sum += f4(n);
    sum += f5(n);
    sum += f6(n);
    sum += f7(n);
    sum += f8(n);
    sum += f9(n);
    sum += f10(n);
    return sum;
}
```