

1. Fill in the former parameters and required attributes.

`private static int Test ( int a, int b, int c )`

```
{
    int t1, t2, t3;        // local variable
    t1 = a + b;
    t2 = a - b;
    t3 = 2 * (t1 + t2);
    c = t1 + t2 + t3;
    return c;
}
```

2. What are the elements of the array *test1* after the following segment of code is executed.

```
double[] test1 = {3, 5, 4, 3, 1, 7};
DoChange(test1, test1.length);
test1: 1.5, 2.5, 2.0, 1.5, 0.5, 3.5
```

The definition of above void method *DoChange* is as follows.

```
private static void DoChange(double[] test, int size)
{
    for(int i=0; i<size; i++)
        test[i] *= 0.5;
}
```

3. What is the output after the following segment of code is executed. `x = 3.833333`

```
double[] test1 = {3, 5, 4, 3, 1, 7};
double x = DoCal(test1, test1.length);
system.out.println("x = " + x);
```

The definition of above value-returning method *DoCal* is as follows.

```
private static double DoCal(double[] test, int size)
{
    double s = 0;
    for(int i=0; i<size; i++)
        s += test[i];
    return s/size;
}
```

4. Write a method *Sum* that receives a double array *myArrays* and its size (# of elements in *myArrays*), then sums up all components of the array.

```
private static double Sum(double[] myArrays, int size)
{
    double s = 0;
    for(int i=0; i<size; i++)
        s += myArrays[i];
    return s;
}
```