Some Simple Programs

Task 1

Implement the following function:

```
	// calculate the number of occurrences of a given value in an
array
	public static int calcOccurenceNumber(int arr[], int value) {
		int result=0;
		// insert the code here
		return result;
	}
	// find the maximal value of an array
	public static int findMax(int arr[]) {
		int result=0;
		// insert the code here
		return result;
	}
	// calculate the number of occurrences of the maximal value
	public static int calcMaxOccurenceNumber(int arr[]) {
		int result=0;
		// insert the code here
		return result;
	}
```

The functions should be tested like this:

```
	public static void main(String[] args) {
		int[] testArray = {12, 12 -234, 2, 777, 12, 777};
		System.out.println("calculated maximal value is: "+
			findMax(testArray)+" should be 777");
		System.out.println("calculated number of occurences is: "+
			calcOccurenceNumber(testArray, 12)+" should be 3");
		System.out.println("calculated max number of occurences is: "+
			calcMaxOccurenceNumber(testArray)+" should be 2");
....
}
```

Task 2

The English mathematician J. Wallis in 1655 presented the following formula for calculating the number of PI.

$$\prod_{n=1}^{\infty} \frac{(2n)(2n)}{(2n-1)(2n+1)} = \frac{2}{1}\cdot\frac{2}{3}\cdot\frac{4}{3}\cdot\frac{4}{5}\cdot\frac{6}{5}\cdot\frac{6}{7}\cdot\frac{8}{7}\cdot\frac{8}{9}\cdots = \frac{\pi}{2}$$

Implement a function to calculates the number PI, n is a parameter:

```
	// use formula according to Wallies
	public static double calculatePiAccWallies(int iterNo) {
		double result=1.0;
	// insert your code here
…
	}
```

Please do NOT forget that inter arithmetic produces integer results!
Therefore you should convert integer value to double by a cast operator (double) :

```
		double result=4/5;
		System.out.println(result);  // 0
		result=(double)4/5;
		System.out.println(result);  // 0.8
```

Task 3

The Scottish astronomer J. Gregory in 1671 presented the following formula for calculating the number PI.

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} = \frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \cdots = \frac{\pi}{4}$$

Implement a function to calculates the number PI, n is a parameter:

```
	// use formula according to Gregory
	public static double calculatePiAccGregory(int iterNo) {
		double result=0.0;
		int powerMinusOne=1;
	// insert your code here
…
	}
```

Task 4

Test the above defined functions in the following way:

```
		double value;
		int iterNo=4;
		value=calculatePiAccWallies(iterNo);
		System.out.println("Wallies for: "+iterNo+"\t"+value);
		value=calculatePiAccGregory(iterNo);
		System.out.println("Gregory for: "+iterNo+"\t"+value);
		iterNo=20;
```

```
value=calculatePiAccWallies(iterNo);
System.out.println("Wallies for: "+iterNo+"\t"+value);
value=calculatePiAccGregory(iterNo);
System.out.println("Gregory for: "+iterNo+"\t"+value);
```

The result should look like:

```
Proper value of PI: 3.141592653589793
Wallies for: 4  2.9257142857142853
Gregory for: 4  2.8952380952380956
Wallies for: 20 3.1015772634382706
Gregory for: 20 3.09162380666784
```

Andrzej Siemiński