Assignment:1

ENSF 594

Ziad Chemali

Question 1: Fibonacci number

Java Code:

```
import java.util.ArrayList;
import java.util.Scanner;
/**
 * This class calculates the Fibonacci of n that the user choose
 * @author Ziad Chemali
 *
 */
public class Fibonacc {
	///creating variables
ArrayList<Double> fibList;//n
ArrayList<Double> fn;//Fn
Double answer;
/**
 * This is constructor where the
 */
public Fibonacc() {
	fibList=new ArrayList<Double>();
	fn=new ArrayList<Double>();

	fn.add((double) 0);
	fn.add((double) 1);
	fibList.add((double) 0);
	fibList.add((double) 1);

}
/**
 * This method use recursive call to find the Fibonacci of n
 * @param n users choice
 * @return Fn=F(n-1)+F(n-2)
 */
public double getF(int n) {


	for (int i=0 ;i<fibList.size();i++)
	{
		if (n == fibList.get(i)) {

			return fn.get(n);
		}}

		answer= (double) (getF(n-1)+getF(n-2));
		fibList.add((double) n);//adding n to array
		fn.add(answer);//adding Fn to array
```

```
		return answer;

	}

	public static void main(String[] args) {
		double time=System.currentTimeMillis();
		Fibonacc fib =new Fibonacc();
		boolean check=true;
		while (check) {
			Scanner scanner =new Scanner(System.in);
			System.out.println("Enter n between 0 and 45");
			int n=scanner.nextInt();
			if (n>=0 && n<=45) {
				System.out.println("Fibonacci of "+n+" is: "+fib.getF(n));
			}
			else {
				System.out.println("Choose n between 0 and 45");
			}
			System.out.println("Do you want to calculate another n Yes/No");
			String answr;
			answr=scanner.next();
			if (!answr.equalsIgnoreCase("Yes")) {
				check=false;
			}

		}




	}

}
```

Result:

```
Enter n between 0 and 45
22
Fibonacci of 22 is: 17711.0
Do you want to calculate another n Yes/No
yes
Enter n between 0 and 45
33
Fibonacci of 33 is: 3524578.0
Do you want to calculate another n Yes/No
yes
Enter n between 0 and 45
999
Choose n between 0 and 45
Do you want to calculate another n Yes/No
no
```

Question 2: GCD

Java Code:

```
import java.util.Scanner;
/**
 * This class calculates the GCD of two numbers using efficient algorithm
 * @author ziad chemali
 *
 */
public class GCD {
	/**
	 * This method divide a/b if remainder is zero then return b
	 * else
	 * recursive the method with b/remainder
	 * @param a numerator
	 * @param b denominator
	 * @return  denominator if remainder of a/b is zero
	 */
	public static int getGCD(int a,int b)
	{
		if(a%b==0)
			return b;

		return getGCD(b,a%b);

	}

	public static void main(String[] args) {
		int a,b;
		Scanner scanner =new Scanner(System.in);
		boolean check=true;
		while (check) {
			System.out.println("Enter two Integers a,b between 1 and
2*10^9");
			a=scanner.nextInt();
			b=scanner.nextInt();
			if(a>=1 && a<=2*Math.pow(10, 9) || (b>=1 && b<=2*Math.pow(10,
9))) {
				System.out.println("GCD for a and b is :
"+GCD.getGCD(a,b));
			}
			else {
				System.out.println("Please enter a valid a,b numbers
berween 1 and 2 *10^9");
			}
			System.out.println("Do you want to continue?");
			String ans;
			ans=scanner.next();
			if(!ans.equalsIgnoreCase("Yes"))
			{
				check=false;
			}

		}
```

```
        }
}
```

Result:

```
Enter two Integers a,b between 1 and 2*10^9
222
3
GCD for a and b is : 3
Do you want to continue?
yes
Enter two Integers a,b between 1 and 2*10^9
11112
3332
GCD for a and b is : 4
Do you want to continue?
yes
Enter two Integers a,b between 1 and 2*10^9
28851538
1183019
GCD for a and b is : 17657
Do you want to continue?
no
```