G51PRG: Introduction to Programming Second semester

Lecture 1

Natasha Alechina
School of Computer Science & IT
nza@cs.nott.ac.uk

Plan of the lecture

- About the course (this semester)
- A little bit of revision
- Teaching evaluation (last semester!)

Running of the course this semester

- Lecturer: Natasha Alechina (nza)
- Tutors and demonstrators - hopefully the same
- Lab allocation: the same (will try to move people from C11 on Thursdays into A32 on Tuesdays).
- Exam (for the whole course): worth 40% of the mark
- Last semester mark: 50%
- So programming exercises this semester only worth 10%
- No point skipping the exercises: can't learn to program without doing some programming.
- Exam easy if understood the exercises.

This week

- No labs
- Mostly revision lectures (some new things in the Friday lecture)
- If you had difficulties with Java last semester, read course notes or textbook, ask your tutor or friends to explain solutions to exercises. I'll try to organise remedial classes if there is enough interest.
- First exercise opens next week on Monday
- Deadline the 17th of February

Plan of the (rest of the) course

- Classes and objects
- Inheritance (extending classes)
- Some useful data structures
- Exceptions
- Input and output in more detail
- A bit of networking
- A bit of multithreading
- After Easter: graphical user interfaces and applets

Revision

- All Java code is inside some class…
- main() method is executed; it may call other methods
- how to call methods
- access modifiers
- constructors

- I assume that you remember basic types of Java, if statements, loops and so on. If not get a textbook!!!

Example 1

```
class  SomeCode {
   public static void echo(String s){
      System.out.println(s);
   }
   public static void main(String[] args){
      if (args.length > 0) {
         echo(args[0]);
      }
   }
}
```


Example 2

```
class  Point {
 int x,y;
 public Point(int x, int y){
      this.x = x;
      this.y = y;
 }
 public print() {
   System.out.println("x is "+ x +", y is " + y);
 }
 public static void main(String[] args){
   Point p = new Point(4,5);
   p.print();
 }}
```


Next lecture

- Classes and objects
- Constructors
- Passing by reference or by value
- Comparing and copying objects