DEPARTMENT OF PHYSICS & ASTRONOMY
PHAS3459 Final Examination
13:30 – 16:30: 20th January 2016

Please read the exam guidelines, rules, instructions and marking criteria at
`https://moodle.ucl.ac.uk/mod/wiki/view.php?pageid=15228`
(linked from the *Examinations and coursework* page).

This exam is worth 50% of your final mark for the course. The duration of the exam is 3 hours.

The Java source code of your solution to the programming exercise should be uploaded using Moodle under the section headed "Exam II".

Each class should be uploaded as a separate file. Your classes should be created in a package called "`exam2`". You must upload all your classes used in your solution, including any you have copied or imported from earlier coursework modules. The code you upload **must be self-contained**: the marker must be able to compile and run it using only the classes uploaded and the Java API. If you use your own classes from earlier modules, make sure you copy them into the `exam2` package and upload them along with any new classes you create during the exam.

You are advised to read the entire exam paper before starting work.

In this examination you will process some data from a fictional experiment to measure the speed of a beam of particles. Bunches of particles are emitted at intervals from a source, and each bunch gives rise to an electrical pulse in any detector that it hits. This pulse is recorded as a series of voltages at intervals of 1 nanosecond, and you will process this raw data to find the *arrival time* and *amplitude* of each pulse.

The input data files are provided in the following web directory: `http://www.hep.ucl.ac.uk/undergrad/3459/exam-data/2015-16/`.

Each line of the file `detectors.txt` contains the following information about one detector:

- an identifier for the detector;
- the distance of the detector from the particle source, measured in metres.

Each line of the file `signals.txt` contains data representing a single pulse from one detector. A line starts with the identifier for the detector that the signal comes from, followed by a series of numbers separated by white space, representing the voltage (in volts) at intervals of 1 nanosecond. The first number corresponds to the time when the particle bunch was emitted from the source.

Part 1: 20/50 marks

The *amplitude* of a pulse is defined as the maximum voltage at any time during the pulse, i.e. the largest number in the series. The *arrival time* can be measured in various ways: in this part of the exercise it is taken to be the first time at which this maximum voltage occurs, relative to the time the pulse was emitted. So if the first value is the largest, the arrival time is zero; if the second value is the largest, the arrival time is 1 ns, and so on.

Write a program to do the following:

- Read the data from the files detailed above, and store them in one or more appropriate data structures.
- Calculate and print:
  - the total number of pulses;
  - the mean amplitude of the pulses.
- For each detector that occurs in the data, calculate and print:
  - the number of signals from this detector;
  - the mean amplitude of the pulses from this detector;
  - the mean arrival time of the pulses from this detector;
  - the speed of the particles, calculated from the mean arrival time and the distance of the detector from the source.

Part 2: 20/50 marks

In order to carry out a more precise analysis, the researchers try using different methods to calculate the arrival time of each pulse.

- Create an interface to represent a method of calculating the arrival time of a pulse.
- Create two implementations of this interface:
  - one that finds the time of the maximum recorded voltage, as in part 1;
  - another that takes the time of the first voltage above a given threshold.
- Use the second implementation to recalculate the speed of the particles using each detector, with a threshold of 1.0 V.

Part 3: 10/50 marks

Write a program to find out and report for which detector the difference between the arrival times calculated using the two methods is greatest.

END OF PAPER