

# University of Hull

## Department of Computer Science

### Playing with Python

Vsn. 1.0 Rob Miles 2014

### Fun with Cricket Scores

#### Processing Cricket Player Scores

This week you are going to perform some simple processing on the scores produced by a team after a cricket match. We are going to start with something simple, and then add extra behaviours as we go along over the next few weeks.

#### Getting Inputs

For each batsman the following information will be entered:

- score achieved in runs

Every member of the team must bat, so 11 scores will be entered into the program. The customer is insisting that the program does not accept invalid inputs. You have been told:

- The scores will be entered at the start of the program. There are eleven scores.
- The lowest score which can be achieved by a batsman is 0. This is also called a “duck”.
- A score of over 100 runs is called a century, and regarded as a good thing in cricket circles.
- The highest score which the program should accept is 500, i.e. if the user tries to enter a score which is greater than this value it must be rejected.
- The average score is the total innings score divided by 11.
- If there are any other errors you can think of, your program should handle them too.

For example (this is just a sample format, you can arrange the input out how you like):

```
150
0
2500
Score too large. Please enter a score between 0 and 500
25
```

This means that the first batsman scored 150 runs, the next scored 0 runs (a duck) and the third scored 25 runs. One invalid score was rejected.

#### Producing Team Statistics

The team manager wants the following statistics from the scores data:

- Total score of the innings
- Number of players who scored a duck
- Number of players who scored a century or more
- Highest score
- Lowest score
- Average score in the innings

### ***Reading in the numbers***

You should use the number reading methods you have created in earlier weeks to read the numbers in and reject invalid ones. The Cinema Entry program is well worth a look for this. If you know how to use methods, then you can create a method that will read the values in for you.

### ***Doing the calculation***

You don't need to store the scores to make the program work. You can store a running total, along with counters for the number of ducks and number of centuries. You can also store the highest and lowest seen so far and update these each time round the loop that processes entries.

### **Advanced: Storing the Scores**

If you have used lists in Python you can think about storing the scores in a list. This will make it easier to process as you can make a for loop that works through it. You could also think about sorting the list, perhaps using "Bubble sort" that works through the values