Part 2

Each pupil auditioning for a place in the youth orchestra will perform two pieces of music. Each performance will be awarded a score out of 50. Pupils aged 12 to 14 who achieve a total score greater than 70 can join the junior orchestra; pupils aged 15 to 17 who achieve a total score greater than 70 can join the senior orchestra. Today Samantha is visiting Westwood Academy to audition four musicians. She requires software to help her with this task.

The system requires the following inputs:

- the name of each pupil
- the age of each pupil
- the score for the first performance
- the score for the second performance.

The output from the program should display each pupil's name, age, total score and a decision indicating whether the pupil is accepted to the junior or senior orchestra, or if the pupil is declined.

An example is provided below

```
Pupil Name          Pupil Age     Total Score    Decision

C Adkins               16             73         Accepted to senior orchestra
J Brown                13             47         Declined
I Shafiq               12             77         Accepted to junior orchestra
```

Your task is to create software for this system.

The top level algorithm is shown below. Step 5 and part of step 7 have been refined.

Pseudocode

MAIN STEPS

```
1.      Loop for each pupil
2.          Get pupil name
3.          Get a valid age
4.          Get two valid scores
5.          Calculate total score
6.      Loop until no more pupils
7.      Display results and decision
```

REFINEMENTS

```
5.      Calculate total score
5.1     Total score = first score + second score


7.      Display results and decision
7.1     Display Headings
7.2     Loop
7.3         Display pupil name
7.4         Display pupil age
7.5         Display total score
7.6         Decide if pupil will be accepted or declined and display decision
7.7     End loop
```

<table>
<tr><th colspan="2">Tasks</th><th>Evidence required</th></tr>
<tr><td>1</td><td>Refine the following parts of the algorithm:
• Get valid age (step 3)
• Get two valid scores (step 4)
• Decide if pupil will be accepted or declined (step 7.6)
(NOTE: <i>all refinements must include an algorithm and not simply use a feature of an event-driven language.</i>)</td><td>Pseudocode for steps 3, 4 and 7.6</td></tr>
<tr><td>2</td><td>Create a program that matches the refined algorithm.</td><td>Listing of program</td></tr>
<tr><td>3</td><td>Copy and complete the test table below.
<table>
<tr><th>Pupil Name</th><th>Pupil Age</th><th>First score</th><th>Second Score</th><th>Total Score</th><th>Expected Decision</th><th>Actual Decision</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>C Adkins</td><td>16</td><td>32</td><td>41</td><td></td><td></td><td></td></tr>
<tr><td>J Brown</td><td>13</td><td>25</td><td>22</td><td></td><td></td><td></td></tr>
<tr><td>I Shafiq</td><td>12</td><td>31</td><td>46</td><td></td><td></td><td></td></tr>
<tr><td>G Kerr</td><td>15</td><td>32</td><td>38</td><td></td><td></td><td></td></tr>
</table>
</td><td>Set of test data</td></tr>
<tr><td>4</td><td>Test your program using the test data.</td><td>Printed output</td></tr>
</table>