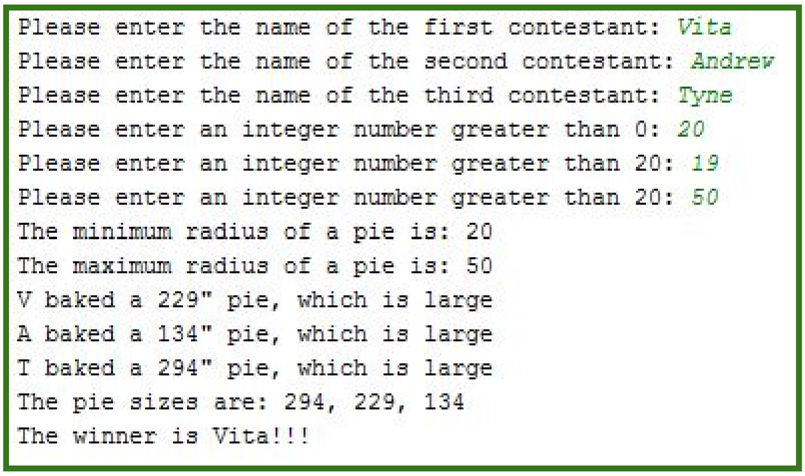
PGCert IT: Programming for Industry

Additional Practice 01

Question One: Pie Baking Competition Game

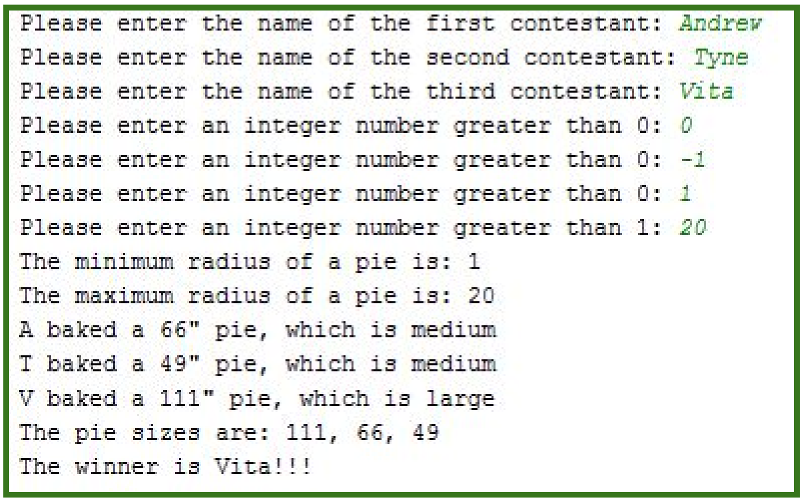
Complete the `GoodPieBakingCompetition` class provided to you (the files are in the `ictgradschool.industry.practice01.q1` program folder) in order to create a simple program that allows three users to play a pie baking competition game. The program first asks each contestant to enter their first name. The program then asks contestants to enter two integer numbers. The first number must be greater than 0 and the second number must be greater than the first number. The first number represents the minimum pie radius and the second number represents the maximum pie radius. The program then randomly generates three pie sizes, one for each contestant. Each pie size is calculated based on the circumference of the pie. The program then displays the pie size for each contestant with their initials and the pie size category. It also displays the three pie sizes in descending order. Finally, the program determines the winner of the good pie competition.

Here is an example output of the program:



```
Please enter the name of the first contestant: Vita
Please enter the name of the second contestant: Andrew
Please enter the name of the third contestant: Tyne
Please enter an integer number greater than 0: 20
Please enter an integer number greater than 20: 19
Please enter an integer number greater than 20: 50
The minimum radius of a pie is: 20
The maximum radius of a pie is: 50
V baked a 229" pie, which is large
A baked a 134" pie, which is large
T baked a 294" pie, which is large
The pie sizes are: 294, 229, 134
The winner is Vita!!!
```

Here is another example output of the program:



```
Please enter the name of the first contestant: Andrew
Please enter the name of the second contestant: Tyne
Please enter the name of the third contestant: Vita
Please enter an integer number greater than 0: 0
Please enter an integer number greater than 0: -1
Please enter an integer number greater than 0: 1
Please enter an integer number greater than 1: 20
The minimum radius of a pie is: 1
The maximum radius of a pie is: 20
A baked a 66" pie, which is medium
T baked a 49" pie, which is medium
V baked a 111" pie, which is large
The pie sizes are: 111, 66, 49
The winner is Vita!!!
```

The program is to be written so that each task is in a **separate method**. You need to complete the following **private** methods, one method for each of the following tasks:

A. Get a name from the user input.

B. Get an integer from the user. If the integer is less than or equal to the lower bound, then ask the user to try again.

C. Print the minimum and maximum radius from the given values.

D. Generate a random number between the given ranges and calculate the circumference using the random number as the radius. Return the value as an integer.

$$C = 2\pi r$$

Where *C* is the circumference and *r* is the radius.

E. Determine the pie size category based on the given value. Return “extra large” if the size is bigger than 500; “large” for size bigger than 100; “medium” for the size bigger than 40; “small” for size bigger than 10; otherwise return “extra small”.

F. Get the contestant initial from the given contestant name.

G. Print the following message on console “[initial] baked a [size] pie, which is [pie category]. **Hint:** Don’t forget to use the methods created previously.

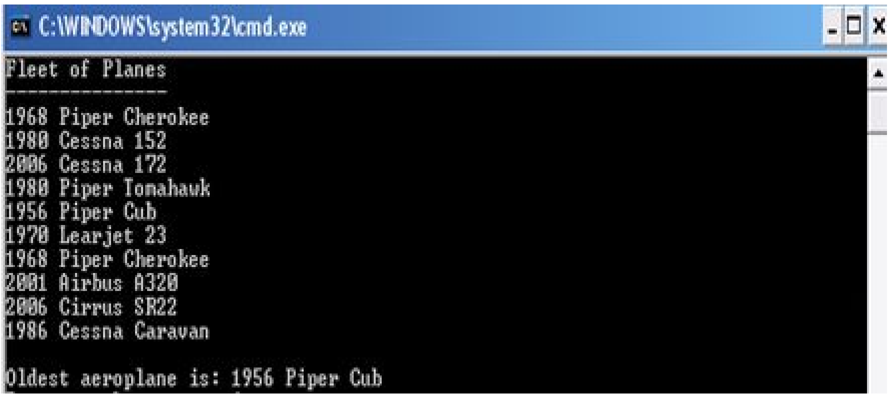
H. Print the given values in descending order. **Note:** Do **not** use conditionals to answer this question.

I. Generate a random number between 1 - 100 and return the remainder of the random number after dividing by 3 as the winner number.

J. Get the winner of the pie competition, which is determined by the winner number. If the winner number is 0, the winner is contestant1. If the winner number is 1, the winner is contestant2. If the winner number is 2, the winner is contestant3. **Note:** You should use the method created in I. to determine the winner number.

Question Two

Complete the `Flying` program (the files are in the `ictgradschool.industry.practice01.q2` program folder) as per the instructions listed in Steps 1 – 9 below so that it produces the following output:



```
Fleet of Planes
---------------
1968 Piper Cherokee
1980 Cessna 152
2006 Cessna 172
1980 Piper Tomahawk
1956 Piper Cub
1970 Learjet 23
1968 Piper Cherokee
2001 Airbus A320
2006 Cirrus SR22
1986 Cessna Caravan

Oldest aeroplane is: 1956 Piper Cub
```

In the Aeroplane class, do the following:

1. Declare instance variables for `make`, `model`, and `year`.
2. Complete the constructor, which assigns the instance variables appropriately.
3. Complete the `toString()` method. This method returns a string of the form `"[year] [make] [model]"`, as you can see in the test handout.
4. Complete the `isOlderThan()` method, which returns a value indicating whether this `Aeroplane` is older than the given one (i.e. its year is smaller).
5. Complete the `equals()` method. If the object is an `Aeroplane`, return true if its make, model and year are the same as this one's. If the object is not an `Aeroplane`, just return false.

In the `Main` class, do the following:

6. Declare an array of `Aeroplane` objects. Name the array **`planes`**.
7. Construct the **`planes`** array that you declared in Step 6. The **`planes`** array needs to be big enough to hold 10 `Aeroplane` objects.
8. Complete the `printPlanesArray()` method. This method takes an array of `Aeroplane` objects as a parameter and prints all the elements as per the screenshot above. Note that the `toString()` method in the `Aeroplane` class can be called to obtain a `String` containing the instance variables of a particular `Aeroplane`, formatted in the required manner.
9. Complete the `getOldestAeroplane()` method. This method takes an array of `Aeroplane` objects as a parameter and returns a reference to the oldest `Aeroplane`. Note that the `isOlderThan()` method in the `Aeroplane` class can be used to determine if an `Aeroplane` is older than another `Aeroplane`.