Problem 1 `BankLoan.java`

Write a program `BankLoan.java` that computes the bank loan (monthly payment and total loan to be paid after including the total interest) based on a given initial loan amount, yearly interest rate and number of loan years. You should create a method, `monthlyPay`, that computes and **returns** the monthly payment. The method takes three parameters (in the same order): 1) number of loan years; 2) initial loan amount; 3) yearly interest rate, which corresponds to a percentage. The formula to compute the monthly payment is as follows:

$$monthly\ payment = \frac{i \times r}{1-(1+r)^{-n}}$$

r: monthly interest rate
n: number of months
i: initial loan amount

Example: `monthlyPay(20, 15000.0, 10.0)` returns `$144.75324676110134.`

Write another method `bankLoan` that computes and prints the following output (make sure to include the special characters and follow the same exact format; you should pass the proper parameters):

```
Loan period in months: 240
Annual interest rate: 10.0%
Initial loan: $15000.0
Monthly payment: $144.75324676110134
Total loan with interest: $34740.77922266432
```

Problem 2 `RandomFair.java`

Write a method `isRandomFair` that takes three integers a, b and n as parameters (where a < b and n > 0). The method generates n random integers between a and b (both inclusive), and **returns** `true` if the difference between (1) the average of the random integers, and (2) the midpoint of a and b, is less than or equal to 2; it prints `false` otherwise. For example, for a = 3, b = 40, n = 10 and if the generated random integers are: `25 13 8 16 15 25 7 20 6 27,` then the method should print the following:

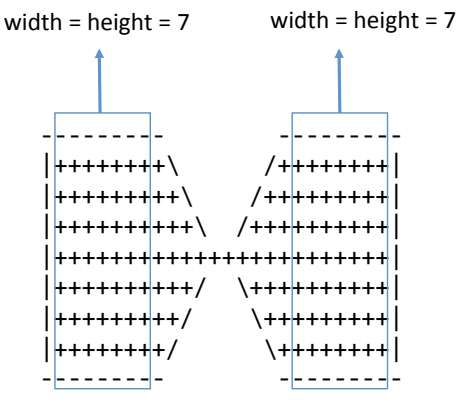
```
For [3, 40] and n = 10 -- random is fair: false
```

In this case, the average of the random integers is equal to 16.2 and the midpoint of a and b is equal to 21.5. The method prints false since the difference between 16.2 and 21.5 is greater than 2.

Write a program **`RandomFair.java`** that calls `isRandomFair` method for a = 1, b = 10 and n varying from 100 to 1000 in 100 increments.

Problem 3 `Bowtie.java`

Write a method, bowtie, that takes an integer height as a parameter. It then uses this parameter to draw and appropriately size the figure below. Assume that the **height (i.e., the number of lines with the '+' character)** is always an odd number greater than 5. The following pictorial illustrates the figure details when height is equal to 7:



Write a Java program, **`Bowtie.java`**, that calls the method for different values of height.

Sample Run 1 (when height = 7)

```
C:> java Bowtie
---------        ---------
|++++++++\      /++++++++|
|+++++++++\    /+++++++++|
|++++++++++\  /++++++++++|
|++++++++++++++++++++++++|
|++++++++++/  \++++++++++|
|+++++++++/    \+++++++++|
|++++++++/      \++++++++|
---------        ---------
```

Sample Run 2 (when height=11)

```
C:> java Bowtie
-------------            -------------
|++++++++++++\          /++++++++++++|
|+++++++++++++\        /+++++++++++++|
|++++++++++++++\      /++++++++++++++|
|+++++++++++++++\    /+++++++++++++++|
|++++++++++++++++\  /++++++++++++++++|
|++++++++++++++++++++++++++++++++++++|
|++++++++++++++++/  \++++++++++++++++|
|+++++++++++++++/    \+++++++++++++++|
|++++++++++++++/      \++++++++++++++|
|+++++++++++++/        \+++++++++++++|
|++++++++++++/          \++++++++++++|
-------------            -------------
```

Sample Run 3 (when height=15)

```
C:> java Bowtie
-----------------                -----------------
|+++++++++++++++\              /+++++++++++++++|
|++++++++++++++++\            /++++++++++++++++|
|+++++++++++++++++\          /+++++++++++++++++|
|++++++++++++++++++\        /++++++++++++++++++|
|+++++++++++++++++++\      /+++++++++++++++++++|
|++++++++++++++++++++\    /++++++++++++++++++++|
|+++++++++++++++++++++\  /+++++++++++++++++++++|
|++++++++++++++++++++++++++++++++++++++++++++++|
|+++++++++++++++++++++/  \+++++++++++++++++++++|
|++++++++++++++++++++/    \++++++++++++++++++++|
|+++++++++++++++++++/      \+++++++++++++++++++|
|++++++++++++++++++/        \++++++++++++++++++|
|+++++++++++++++++/          \+++++++++++++++++|
|++++++++++++++++/            \++++++++++++++++|
|+++++++++++++++/              \+++++++++++++++|
-----------------                -----------------
```

Submission Instructions and Guidlines

- Your submission must consist of a zip archive that contains three .java files only (called **`BankLoan.java, RandomFair.java, Bowtie.java`**). No additional files should exist in the .zip archive.
- Give meaningful names to methods and variables in your code.
- Include a comment at the beginning of your program with basic information and a description of the program and include a comment at the start of each method.
- The name of the zip file must adhere to the following naming convention `asst5_<netid>`, where *<netid>* stands for you AUBnet user name. These zipped files will be processed automatically so please make sure you use this naming convention. The single zipped file must be uploaded to Moodle.