**Ying Pei A13595990**

**public static String binaryString(int n)**

Base case: if the integer n divide by 2 is equal to 0, return n. This is because n can only be 0 or 1 to be equal to zero.

Recursive steps: else return the method by this time with n / 2 as the input value plus the remainder or n/2.

**public static String generateRandomMnemonic(String digits)**

Base case: if the digits is zero, return empty string.

Recursive steps: else we use the helper method to convert the last digits in the string to get a random character. Then we return the method but this time without the last digits and the random character.

**public static boolean isSubSetSum(ArrayList<Integer> set, int targetNumber)**

Base case: if the targetNumber is zero, return true. If the first number is equal to the targetNumber, return true.

Recursive steps: else remove the first number and return the same method or the targetNumber - the first number.

**public static void spiral(Turtle turtle, double initialLength, int angle, double multiplier)**

Base case: if the initialLength equals 0, end the program.

Recursive steps: turtle go forward initialLength and turn angle. Then return the same method but initialLength \* mulplier as the input value.

**public static void tree(Turtle turtle, int trunkLength, int height)**

Base case: if the height equals 0, end the program.

Recursive steps: turtle moves forward at trunkLength. Then it turns -45 degrees and runs the method **tree(turtle, (int)(trunkLength \* 0.5), height - 1).** Then it asks the turtle to turn 90 degrees and runs the method **tree(turtle, (int)(trunkLength \* 0.5), height - 1)**. Then it asks the turtle to go backward at trunkLength.

**public static double humanPyramidWeight(ArrayList<ArrayList<Double>> weights,**

**int level, int offset)**

Base case: if the level equals zero, return the person’s weight. Else if offset is greater than level, return 0.

Recursive steps: if the offset is equal level, return the methods 0.5 \* humanPyramidWeight(weights, level - 1, offset - 1) and the person’s own weight.

Else if offset is zero, returns the method 0.5 \* **humanPyramidWeight(weights, level - 1, offset)** and the person’s own weight. Else, return the average of return **(humanPyramidWeight(weights, level - 1, offset -1)** and **humanPyramidWeight(weights, level - 1, offset +1))** and the person’s own weight.