**Java program:** Prob01.java

**Input File:** Prob01.in.txt

**Output:** Your output needs to be directed to stdout (i.e., using System.out.println())

**Introduction**

Are you tired of traditional game boards like checker boards or chess boards? Us too! We would like to invent a game that is played on a board where every square is the same, and we need your help to construct the game board. You’ll need to write a program to help you, of course.

Write a program which displays an N x N square board made using the pound sign (#). For example N=5 would produce:

# # # # #

# # # # #

# # # # #

# # # # #

# # # # #

**Program Input**

The first line of the file Prob01.in.txt will contain a positive integer T denoting the number of test cases that follow. Each test case will have the following input:

* A single integer N

**Example Input:**

2

5

3

**Program Output**

For each test case, your program should output the NxN board. Put spaces between your pound signs, but not at the beginning or the end of the line.

**Example Output:**

# # # # #

# # # # #

# # # # #

# # # # #

# # # # #

# # #

# # #

# # #

**Java program:** Prob02.java

**Input File:** Prob02.in.txt

**Output:** Your output needs to be directed to stdout (i.e., using System.out.println())

**Introduction**

Some people really like coins. You happen to be one of those people! In a world filled with electronic payments and banking from your cell phone, you prefer to deal exclusively in cash – specifically coins that are generally available (no silver dollars or half dollars for you). However, you are also practical. I mean, paying for your lunch in pennies? That’s just silly. In order to strike the right balance of quirky and practical, you need to be able to pay for things using the fewest number of coins possible – and your program will help you do just that.

**Program Input**

The first line of the file Prob02.in.txt will contain a positive integer T denoting the number of test cases that follow. Each test case will have the following input:

* A single line denoting the dollar amount to be converted.

**Example Input:**

4

$3.87

$2.74

$14.84

$0.76

**Program Output**

For each dollar amount given, have your program calculate the fewest number of coins necessary to arrive at that dollar amount. Your program will output the following:

* The first line should contain the dollar amount exactly as it appears in the input.
* The next line should contain the amount of quarters in the following format:
  + Quarters=<amount of quarters calculated>
* The next line should contain the amount of dimes in the following format:
  + Dimes=<amount of dimes calculated>
* The next line should contain the amount of nickels in the following format:
  + Nickels=<amount of nickels calculated>
* The next line should contain the amount of pennies in the following format:
  + Pennies=<amount of pennies calculated>
* NOTE: If a coin was not used, still include that coin in the output with a “0” amount (i.e. Pennies=0).

**Example Output:**

$3.87

Quarters=15

Dimes=1

Nickels=0

Pennies=2

$2.74

Quarters=10

Dimes=2

Nickels=0

Pennies=4

$14.84

Quarters=59

Dimes=0

Nickels=1

Pennies=4

$0.76

Quarters=3

Dimes=0

Nickels=0

Pennies=1