Q1. Print Prime number series in an hourglass shape.

The user will enter the number of rows and the output should contain those many rows printing the prime number series and a mirror image of the hourglass with the text ANAMI side by side (exactly as shown in the below example).

Example:

Input: 7 (Number of Rows)

Output:

```
2+3+5+7   ANAMIAN
  11+13     ANAMI
   17+       ANA
    1         A
   9+2       ANA
  3+29+     ANAMI
31+37+4   ANAMIAN
```

NOTE: To form a perfect hourglass, split the numbers of the prime number series to next row and for the last number of the prime number series, discard any remaining digits after the split (In example: "19" was split to place "9" into the next row to form an hourglass and "41" is split to place "4" in the last row and remaining digit '1' is discarded). For the ANAMI text, repeat the text (ANAMI) to make a perfect diamond shape for large rows. **TWO SHAPES MUST BE SIDE BY SIDE.**

Q2. Implement Quicksort algorithm to sort the below problem.

- The user will enter the student names
- The user will enter the subjects (as many subjects as he/she wants)
- The user will enter the marks for all students for each subject
- You will total the marks for each student
- You will store them in a dynamic list. The user should be able to add as many names, subjects and marks as he/she wants.
- You should be able to sort the names based on their total marks they received in descending order and print the output as shown below.

Example:

Input:

Names = {'Nicole', 'Sam', 'Ryan'}   Subject = {'English', 'Math', 'Science'}
Marks = Nicole: {30, 50, 40}, Sam: {50, 70, 50}, Ryan: {40, 40, 30}

Output:

```
Rank  | Name      | English   | Math      | Science   | Total Marks
-----------------------------------------------------------------------------------------------------
1     | Sam       | 50        | 70        | 50        | 170
2     | Nicole    | 30        | 50        | 40        | 120
3     | Ryan      | 40        | 40        | 30        | 110
```

NOTE: You must implement the quicksort algorithm to sort the list. DO NOT USE ANY BUILT IN FUNCTION TO SORT THE LIST.