15CSE180 Computer Programming Lab
Periodical Lab - 1
(50 Marks)
Set – 1

1. You are provided with an array of `n` responses (whose values range from `1` to `9`) to a particular survey. Write a program to draw a histogram of the survey response results.

   For example given the following array with 99 responses

```
{ 6, 7, 8, 9, 8, 7, 8, 9, 8, 9,
  7, 8, 9, 5, 9, 8, 7, 8, 7, 8,
  6, 7, 8, 9, 3, 9, 8, 7, 8, 7,
  7, 8, 9, 8, 9, 8, 9, 7, 8, 9,
  6, 7, 8, 7, 8, 7, 9, 8, 9, 2,
  7, 8, 9, 8, 9, 8, 9, 7, 5, 3,
  5, 6, 7, 2, 5, 3, 9, 4, 6, 4,
  7, 8, 9, 6, 8, 7, 8, 9, 7, 8,
  7, 4, 4, 2, 5, 3, 8, 7, 5, 6,
  4, 5, 6, 1, 6, 5, 7, 8, 7 }
```

   your program should produce

```
Response        Frequency       Histogram
1               1               *
2               3               ***
3               4               ****
4               5               *****
5               8               ********
6               9               *********
7               23              ***********************
8               27              ***************************
9               19              *******************
```

   You can check the correctness of your output by validating that the sum of frequencies should be equal to the size of the input survey response array.

2. Given an array of non-negative integers, write a program to find the minimum number of elements such that their sum should be greater than the sum of the rest of the elements of the array.

   Given `{3, 1, 7, 1}` the output should be `1` element (i.e. `{7}`) since 7 is greater than the sum of the rest of the elements i.e `3+1+1 = 5`

   Given `{2, 1, 2}` the output should be `2` elements (i.e. `{2,1}` or `{1,2}` and `{1,2}`) since `3` and `4` is greater than the sum of the rest of the elements i.e `2` and `1` respectively.

15CSE180 Computer Programming Laboratory
Periodical Lab – 1
(50 Marks)
Set – 2

1. Write a program to show that the frequency of numbers that you get while rolling a six-sided die, say for 6, 000, 000 (6 million) times, occur approximately with *equal likelihood*!!
   (Hint: The element of chance can be introduced using library function `rand()` from the `<stdlib.h>` header. The rand function generates an integer between `0` and `RAND_MAX` (a symbolic constant defined in the `<stdlib.h>` header which is a very large number). So producing integers in the range `0` to `5` requires using modulus operator!!)

   When you run your program a typical output should look like the following

```
Face      Frequency
1         999702
2         1000823
3         999378
4         998898
5         1000777
6         1000422
```

   You can check the correctness by summing the frequency column which should result in 6, 000, 000!!!

2. Given an array of size `n` of integers in range from `1` to `n`, write a program to find the inverse permutation of that array. An inverse permutation is a permutation which you will get by inserting position of an element at the position specified by the element value in the array.

   For example given an array `{2, 3, 4, 5, 1}` the output should be `{5, 1, 2, 3, 4}` because of the following conversions (assuming that positions start from 1)
   - value 2 in position 1 becomes value 1 in position 2
   - value 3 in position 2 becomes value 2 in position 3
   - value 4 in position 3 becomes value 3 in position 4
   - value 5 in position 4 becomes value 4 in position 5
   - value 1 in position 5 becomes value 5 in position 1