Camel

```
/*
Camels Transportation Problem
Problem Description
A man has some camels, and he needs to move all of them to the opposite
end of a desert.
 Each camel has a specific cost associated with traveling from one end to
the other.

The man can take two camels at a time to the opposite end, and the cost of
the trip will be the maximum cost of the two camels.
After reaching the opposite end, the man returns with one camel, and the
cost of returning is the cost of that particular camel.
The goal is to find the minimum total cost to move all the camels to the
opposite end.
Example
Input
2                // Number of test cases
4                // Number of camels in the first test case
1 2 8 9          // Costs of camels in the first test case
6                // Number of camels in the second test case
14 45 73 86 95 98 // Costs of camels in the second test case
input:
2
4
1 2 8 9
6
14 45 73 86 95 98
Output
16
434
*/
```

Crows Pot

```
/*
https://codereview.stackexchange.com/questions/136165/the-thirsty-crow

There are N pots. Every pot has some water in it. They may be partially
filled. So there is a Overflow Number O associated with every pot which
tell how many minimum stone pieces are require for that pot to overflow.
So if for a pot O-value is 5 it means minimum 5 stone pieces should be put
in that pot to make it overflow.

Initially a crow watched those pots and by seeing the water level he
anticipated O-value correctly for every pot (that is he knew O1 to On).
But when he came back in evening he found that every pot is painted from
outside and he is not able to know which pot has what O-value.
The Crow wants some K pots to overflow so that he can serve his child
appropriately. For overflowing the pots he needs to search for the stones
in forest (assume that every stone has same size).

He wants to use minimum number of stones to overflow the K pots. But he
doesn't know now which pot has what O-value. So the task is to find out
the minimum number of stones that the crow requires to make the K pots
overflow in the worst case.

Input Specification:

A array O corresponding to O-value of N pots {O1, O2, ..... , On}
Number of pots
K -value ( number of pots which the crow wants to overflow)
Output Specification:

Minimum number of stones required to make K pots overflow in worst case or
-1 if input is invalid.

Example:

Let's say there are two pots:

Pot 1 has O value of 5 , O1= 5
Pot 2 has O value of 58, O2= 58
```

```
Let's say the crow wants to make one of the pots overflow.
If he knows which pot has what O-value he would simply search for 5 stones
and put them in pot 1 to make it overflow. But in a real case he doesn't
know which pot has what O-value so just 5 stones may not always work.
However he does know that one pot has O-value 5 and other has 58.
So even in the worst case he can make one of the pot overflow just by
using 10 stones.
He would put 5 stones in one pot if it doesn't overflow he would try the
remaining 5 in the other pot which would definitely overflow because one
of the pot has O-value of 5.
So the answer for above question is minimum 10 stones even in worst case.
*/
```

Doctor’s Probability

```
/*
https://www.geeksforgeeks.org/samsung-interview-experience-set-39-campus-r
-d-noida/
https://www.careercup.com/page?pid=samsung-interview-questions

A Doctor travels from a division to other division where divisions are
connected like a graph(directed graph) and
the edge weights are the probabilities of the doctor going from that
division to other connected division but the
doctor stays 10mins at each division now there will be given time and had
to find the division in which he will be
staying by that time and is determined by finding division which has high
probability.

Input is number of test cases followed by the number of nodes, edges, time
after which we need to find the division
in which he will be there, the edges starting point, end point,
probability.

Note: If he reaches a point where there are no further nodes then he
leaves the lab after 10 mins and the traveling
time is not considered and during that 10min at 10th min he will be in
next division, so be careful

Sample Input
```

```
2
6 10 40
1 2 0.3
1 3 0.7
3 3 0.2
3 4 0.8
2 4 1
4 5 0.9
4 4 0.1
5 6 1.0
6 3 0.5
6 6 0.5

6 10 10
1 2 0.3
1 3 0.7
3 3 0.2
3 4 0.8
2 4 1
4 5 0.9
4 4 0.1
5 6 1.0
6 3 0.5
6 6 0.5

Sample Output
6 0.774000
3 0.700000
*/
```

Equation Solve

```
/*
Q- We have an equation k= (an + bnlog(base2)n + cn^3) , k is achieved by
enumeration on this equation.
We have to find which value of n satisfy above eqaution.

if no such value of n then return 0.
(Constraints are for Binary search) .
input a ,b ,c ,k .
```

```
logbase2 function is already implemented in IDE.
P.S Overall easy but tricky -- we have to find optimal search space
according to 'c'.
input:
1 0 1 9
output:
0
input:
2 0 1 12
output:
2
*/
```

Frog Jump

```
/*
Frog Jump Problem
Problem Description
Given a 2D matrix where:

1 represents the places where the frog can jump.
0 represents empty spaces.
The frog can:

Move horizontally without any cost (only on 1s).
Move vertically with a cost equal to the number of jumps taken.
Given a source and destination within the matrix, the frog needs to reach
the destination while minimizing the total vertical jump cost.

Input
The matrix size and its elements.
The source and destination coordinates.
Example
Input
5
0 1 1 0 1
1 0 0 1 0
1 0 0 0 1
0 1 0 1 1
1 0 1 1 1
```

```
4 2 1 3
Output
2
*/
```

Laughing Bomb

```
/*
https://www.cnblogs.com/kingshow123/p/practicec1.html

*/
```

Max Pole Height

```
/*
https://www.geeksforgeeks.org/samsung-r-d-noida-question-september-2018

** You can run your code here:
https://leetcode.com/problems/tallest-billboard/


Maximum Pole Problem
Problem Description
You need to place an electronic banner on top of two pillars of equal
height to maximize visibility across the city.
You are given an array of integers representing possible pillar heights.
Your task is to choose two pillars of the same height from this array to
maximize the combined height of the two pillars.

If no two pillars of equal height can be found, return 0.
Input
The first line contains the number of test cases.
For each test case:
The first line contains an integer N, the number of possible pillar
heights.
The second line contains N integers representing the heights of the
pillars.
Example
Input
```

```
1
5
1 2 3 4 6
Output
8



--------------------
The question was like this... You have to place an electronic banner of a company as high as it can be, so that whole the city can view the banner standing on top of TWO PILLERS.
The height of two pillers are to be chosen from given array.. say [1, 2, 3, 4, 6]. We have to maximise the height of the two pillars standing side by side, so that
the pillars are of EQUAL HEIGHT and banner can be placed on top of it. In the above array, (1, 2, 3, 4, 6) we can choose pillars like this, say two pillars as p1 and p2..
Then pillars can be, p1 = 3 unit... Choosing element (3) from array, Similarly p2 = 3 choosing (2 + 1) from array. Since, two pillars are equal, we can put board on it...
 But we have two maximise the height of the pillars, And if we check for other heights, we can see p1 = 6 p2 = 4 + 2 which is greater than 3 ( the previous height)..
 We have to see if we can further maximize the height... Yes it can be 8. I.e. p1 = 6 + 2 = 8. p2 = 4 + 3 + 1 = 8. Both pillars are equal and banner can be placed...
  And since this is the maximum height attainable for two pillars, we print the answer as 8. In case, there is no combination possible, print 0 (zero).
  INPUT : 1 5 1 2 3 4 6 First line is  T  number of test cases to be followed. Second line of input is number of different pillars. Third line of input is
  different available heights of pillars. Note : heights of given pillars can be same .. I.e. array can have same elements repeated. Output. 8
  Simply print the maximum height attainable so that board/ banner can be placed. In case there is no possible combination for placing banner with equal weighted pillars,
  then print 0. Constraint's : some general constraints were given*/
```

Men’s Restroom

```
/*
Men's Restroom Problem
Problem Description
In a restroom with a certain number of stalls, visitors prefer to maximize
their distance from already occupied stalls. The strategy is to occupy the
middle of the longest sequence of unoccupied stalls.

Given the number of stalls, simulate the process of visitors entering the
restroom and occupying stalls according to the described strategy.

Input
The input consists of a single integer representing the number of stalls.
Example
Input
10
Output
_ _ _ _ _ _ _ _ _ _= maxlen=10, lastidx=9, start= 9-10+1=0, mid(0+9)/2=4
_ _ _ _ X _ _ _ _ _ = maxlen = 5, lastIdx=9, start=9-5+1=5, mid(5+9)/2=7
_ _ _ _ X _ _ X _ _= maxlen=4, lastIdx=4, start=4-4+1=1, mid(1+4)/2=2
_ X _ _ X _ _ X _ _
_ X _ _ X _ _ X X _
_ X _ _ X X _ X X _
_ X X _ X X _ X X _
_ X X _ X X _ X X X
_ X X _ X X X X X X
_ X X X X X X X X X
X X X X X X X X X X
Output Format
Each line represents the stalls' status after each visitor occupies a
stall.
Use _ for unoccupied stalls and X for occupied stalls.*/
```

Oil Mines

```
/*
https://www.careercup.com/question?id=5740719907012608
```

```
https://stackoverflow.com/questions/39673898/divide-array-into-k-contiguos
-partitions-such-that-sum-of-maximum-partition-is-m/39675396
http://ideone.com/r60yH4 - Sameer Code
https://www.careercup.com/question?id=5730470801702912
*/

/*
There is an island surrounded by oil mines. You will be given n companies
and m oil mines having values.
You have to distribute the mines to "n" companies in fair manner. Remember
the companies can have oil
mines adjacent to each other and not in between of each others.After
distributing them compute the
difference of oil mines from the company getting highest and company
getting lowest. This number
should be minimum.(then only the distribution can be termed as fair).

Input
2
2 4
6 13 10 2
2 4
6 10 13 2

output
5
1
*/
```

Old Phone Calculator

Omnious Number

Physical Energy

```
/*
Physical Energy Problem
Problem Description
You have a certain amount of energy and need to travel a specific
distance. You can choose from five different speeds to travel, each with
its own energy cost and time taken per kilometer.

Your goal is to minimize the total time required to cover the distance,
ensuring that you do not exceed the available energy.

Given
An initial amount of energy H.
A total distance D to travel.
Five speeds, each with associated:
Cost of traveling 1 km (energy required per km).
Time taken to travel 1 km (time required per km).
Input
An integer H representing the initial amount of energy.
An integer D representing the total distance to be traveled.
A list of five integers representing the energy cost to travel 1 km for
each speed.
A list of five integers representing the time taken to travel 1 km for
each speed.
Example
Input
3000---> limit highest energy
10 ---> distance to travel (km)
4 5 2 3 6 ---> energy cost
200 210 230 235 215 ---> time taken (seconds)
input:
3000
10
4 5 2 3 6
200 210 230 235 215

output 2000

*/
```

Research Center

```
/*
A Research team want to establish a research center in a region where they
found some rare-elements.
They want to make it closest to all the rare-elements as close as possible
so that they can reduce
overall cost of research over there. It is given that all the
rare-element's location is connected
by roads. It is also given that Research Center can only be build on road.
Team decided to assign
this task to a coder. If you feel you have that much potential.

Here is the Task :- Find the shortest of the longest distance of research
center from given locations
of rare-elements.

Locations are given in the matrix cell form where 1 represents roads and 0
no road.
Number of rare-element and their location was also given(number<=5) and
order of square matrix
was less than equal to (20).
*/

/*
For this you have to implement bfs for every position where road exist to
find the distance of
every research center or do Vice-versa. for each position store maximum
distance of all distances
to research center and the compare each maximum distance to find minimum
of them

Input -
t
matrix size   num of rare-element
position of rare-element
matrix elements (0 or 1)

6
5 2
```

```
4 3
3 4
1 1 0 0 0
1 1 0 0 0
1 1 1 1 1
1 1 1 0 1
1 1 1 1 1
8 2
5 6
6 4
1 1 1 1 1 1 0 0
1 1 1 1 1 1 1 0
1 1 0 1 0 1 1 0
1 1 1 1 0 1 1 0
1 1 1 1 1 1 1 0
1 1 1 1 1 1 1 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
10 3
8 2
5 3
7 1
0 0 0 1 1 1 1 1 1 0
1 1 1 1 1 1 1 1 1 0
1 0 0 1 0 0 0 0 1 0
1 1 1 1 1 1 1 1 1 1
1 1 1 1 0 1 0 0 1 1
1 1 1 1 0 1 0 0 1 1
1 1 1 1 0 1 0 0 1 1
1 1 1 1 1 1 1 1 1 1
1 1 1 0 0 1 0 0 1 1
1 1 1 1 1 1 1 1 1 1
15 4
11 15
15 9
1 2
14 3
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 0 1 1 1 1 1 1 1 1 1 1 1 0 1
1 0 1 0 0 0 1 0 0 0 0 1 1 0 1
```

```
1 0 1 0 0 0 1 0 0 0 0 1 1 0 1
1 0 1 1 1 1 1 1 1 1 1 1 1 1 1
1 0 1 0 0 0 1 0 0 0 0 1 1 0 1
1 0 1 0 0 0 1 1 1 1 1 1 1 1 1
1 0 1 0 0 0 1 0 0 0 0 1 1 0 1
1 0 1 0 0 0 1 0 0 0 0 1 1 0 1
1 0 1 0 0 0 1 0 0 0 0 1 1 0 1
1 0 1 0 0 0 1 0 0 0 0 1 1 0 1
1 0 1 0 0 0 1 0 0 0 0 1 1 0 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
0 0 1 0 0 0 1 1 1 1 1 1 1 0 1
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1
20 4
13 6
20 4
1 2
17 16
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0
1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0
1 0 1 0 0 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0
1 0 1 0 0 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0
1 0 1 0 0 0 0 0 0 0 1 0 0 1 1 1 0 0 0 0
1 0 1 0 0 0 0 0 0 0 1 0 0 1 1 1 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 0 1 0 0 0 0 0 0 0 1 0 0 1 1 1 0 0 1 1
1 0 1 0 0 0 0 0 0 0 1 0 0 1 1 1 0 0 1 1
1 0 1 0 0 0 0 0 0 0 1 0 0 1 1 1 0 0 1 1
1 0 1 0 0 0 0 0 0 0 1 0 0 1 1 1 0 0 1 1
1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 1
1 0 1 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 1 1
1 0 1 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 1 1
1 0 1 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
5 2
2 1
3 5
```

```
1 0 1 1 1
1 1 1 0 1
0 1 1 0 1
0 1 0 1 1
1 1 1 0 1

Output -
1
2
2
12
15
4
*/
```

Bipartite Graph Coloring

```
/*
https://www.careercup.com/question?id=5137923582722048
*/

/*
Given a graph print either of the set of the vertices that are colored
with the same color. And if the graph
is not bipartite print "-1". Test cases also included the cases when a
graph is not connected.
input:
6 5
1 2
1 3
2 4
3 5
5 6

output:
colored 1
1 4 5
 colored 0
```

```
2 3 6
*/
```