



MS CONNECT

Weekend Assignment 1

Objectives

- Get comfortable with .NET.
- Start thinking more carefully.
- Solve problems in C#

Reasonable

- Communicating with colleagues about problem problems in English (or some other spoken language).
- Discussing the assignment material with others in order to understand it better.
- Helping a colleagues identify a bug in his or her code, as by viewing, compiling, or running his or her code, even on your own computer.
- Incorporating snippets of code that you find online or elsewhere into your own code, provided that those snippets are not themselves solutions to assigned problems and that you cite the snippets' origins.
- Sending or showing code that you've written to someone, possibly a colleagues, so that he or she might help you identify and fix a bug.

Rules

- Draw the flow chart for each solutions.
- Write the Algorithm for each solution.
- Document your solutions in word file.
- Write proper comments for each line in your source code.
- Document the output of your program.
- Your program should address the problem, there should NOT be any deviations in output.

Question 1

DNA Sequence

Write a program called p2 that will read in a dna sequence for example GTTCAG .

Then the program will search for this dna sequence in a String in the p2 program called dnaDatabase.

It will print out the characters that occur directly before and after this sequence whenever it occurs in the dnaDatabase String.

```
String dnaDatabase =  
"CCTGTATTAGGTTTCAGAAATTCGTTTCAGCAGCAGATTTCGATTAGCTTTACAACAATTCAATA" +  
"AAATAGCTTCGCGCTAAATTTTAACTTTTCTCTGTCGTCGCACAATCGACTTTCTCTGTT" +  
"TTCTTGTTTCAGGGGTTTACCGGAATTGTTTCTGCTGCGATGAGGTATTGCTCGTCAGCCTG" +  
  
"AGGCTGAAAATAAAATCCGTGGTCACACCCAATAAGTTAGAGAGAGTACTTTGACTTGGAG" +  
"CTGGAGGAATTTGACATAGTCGAGTTCAGTTCTTCTCCAAGACGCATCCACGTGAACCGTT" +  
"GTAACATGTTCTGTGCCACACCAAAAAAACTTTCCACGTGAACCGAAAACGAAAGTCTT" +  
"TGGTTTAAATCAATAAGTGCTCTCTTCTCGGAGAGAGAAGGTGGGCTGCTTGCTGCGCAT" +  
"GTACTTTATTAATCAATAACCAACCAAAAAAACTTTCCACGTGTGAACATACTCAA" +  
"AAACGAAGTATTGGTTTATCATAATCTGAAAAGTGCAAAGACGATGATGATGATAGATA" +  
"GGAACCTGAGCAGCCATGTCTGAACCTATAGCGTATTGGTCGTCGTGCGACTAAATTAGGT" +  
"AAAAAGTAGTTCTAAGAGATTTTGATGATTCAATGCAAAGTTCTATTAATCGTTCAATTG";
```

String variable dnaDatabase is a character array.

Input

GTTCAG

If the input string was that shown above and the program was written in C# it would be compiled and run as described below;

Output

G A

C C

T G

A T

To check this in string - GTTCAG

```
CCTGTATTAG GTTCAG AATTC GTTCAG CAGCAGATTTCGATTAGCTTTACAACAATTCAATA  
AAATAGCTTCGCGCTAAATTTTAACTTTTCTCTGTCGTCGCACAATCGACTTTCTCTGTT  
TTCTT GTTCAG GGGTTTACCGGAATTGTTTCTGCTGCGATGAGGTATTGCTCGTCAGCCTG
```

AGGCTGAAAATAAAATCCGTGGTCACACCCAATAAGTTAGAGAGAGTACTTTGACTTGGAG
CTGGAGGAATTTGACATAGTCGA GTTCAG TTCTTCTCCAAGACGCATCCACGTGAACCGTT

you can see below the dna sequence with letters that occur before and after it.

G GTTCAG A

C GTTCAG C

T GTTCAG G

A GTTCAG T

Question 2

You probably know the Fibonacci numbers. They are defined as:

$\text{fib}(0) = 0$

$\text{fib}(1) = 1$

$\text{fib}(n) = \text{fib}(n-1) + \text{fib}(n-2)$ (for $n > 1$)

The Fibonacci numbers grow exponentially, we can only store up to $\text{fib}(47)$ using `int` or $\text{fib}(92)$ using `long` in `C#` or check the appropriate datatype.

Trying to reduce this exponential grow, we decided to introduce this *new* definition of Fibonacci numbers that we call $\text{fib3}(n)$:

$\text{fib3}(0) = 0$

$\text{fib3}(1) = 1$

$\text{fib3}(2) = 1$

$\text{fib3}(n) = \text{fib3}(n-1) + \text{fib3}(n-2) - \text{fib3}(n-3)$ (for $n > 2$)

Your task is, given a list of values, compute the value of $\text{fib3}(n)$ for each of them.

Input

The first line contains an integer `C` with the number of test cases that follow. The next `C` lines contain an integer `N` each.

Limits

$1 \leq C \leq 50$

$0 \leq N \leq 1000000000000000000$ (1018)

Output

For each N, compute fib3(N).

Input Example

7
0
1
2
3
948
93480
56

Output Example

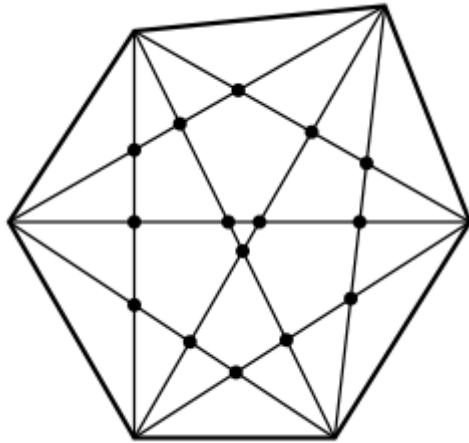
0
1
1
2
474
46740
28

Note that you need to use 64 bits integers to read the input.

Question 3

Consider a convex polygon with N vertices, with the additional property that no three diagonals intersect in a single point. Find the number of intersections between pairs of diagonals in such a polygon.

The image below shows one such polygon with 6 vertices.



Note: a polygon is convex if all of its interior angles are less than 180 degrees.

Input

The first and only line of input contains a single integer N , $3 \leq N \leq 100$.

Output

Output the number of intersections on a single line.

Sample test data

input1

3

output1

0

input2

4

output2

1

input3

6

output3

15

Question 4

Every book has a unique identifier called its ISBN (there are several versions, the latest being called ISBN-13).

Typical ISBNs are:

978-0-470-12870-1

978 0 387 98259 5

An ISBN consists of 13 digits organised as 5 non-empty groups separated from one another by a space or a hyphen.

The first group is either 978 or 979. The final group is a single digit called the checksum. Denoting the sequence of 13 digits by

$x_0x_1x_2 \dots x_{12}$ the checksum (x_{12}) equals

$(10 - (x_0 + 3x_1 + x_2 + 3x_3 + \dots + 3x_{11}) \bmod 10) \bmod 10$

where mod is the arithmetic modulo operator.

Write a program that reads a string from the standard input and display a 1 or 0 as to whether the string is a valid ISBN. 1 being the ISBN number is valid and 0 being that it is not.

The input can be a single ISBN number or a set, s of ISBN numbers.

Input

978-0-306-40615-7

978-1-56619-909-4

978-0-470-12870-1

Output

1

1

1

Question 5

You are given two non-empty strings S and T of equal lengths. S contains the characters '0', '1' and '?', whereas T contains '0' and '1' only. Your task is to convert S into T in minimum number of moves. In each move, you can do one of these changes:

- change a '0' in S to '1'
- change a '?' in S to '0' or '1'
- swap any two characters in S

As an example, suppose S = "01??00" and T = "001010". We can transform S into T in 3 moves:

Initially S = "01??00"

Move 1 – change S[2] to '1'. S becomes "011?00"

Move 2 – change S[3] to '0'. S becomes "011000"

Move 3 – swap S[1] with S[4]. S becomes "001010"

S is now equal to T.

Input

The first line of input is an integer C ($C \leq 200$) that indicates the number of test cases. Each case consists of two lines. The first line is the string S consisting of '0', '1' and '?'. The second line is the string T consisting of '0' and '1'. The lengths of the strings won't be larger than 100.

Output

For each case, output the case number first followed by the minimum number of moves required to convert S into T. If the transition is impossible, output -1 instead. Check the output example for the exact format.

Input Example

```
3
01??00
001010
01
10
110001
000000
```

Output Example

```
Case 1: 3
Case 2: 1
Case 3: -1
```

Question 6

The pirate crew of the Jolly Roger ship have found a hoard of treasure and decide to divide it among themselves. Each pirate has a unique rank and the highest rank pirate gets to suggest how the gold should be divided. The pirates then vote on his plan. His plan is accepted if at least half the crew vote in favour of it. Otherwise, there is a mutiny and that pirate is made walk the plank. The next highest ranked pirate then takes over and the proposal/voting procedure is repeated.

All pirates always act logically, and try to avoid walking the plank while maximising their own share of the gold.

Given that there are G gold coins in the treasure chest and P pirates in the crew, how would you suggest dividing the gold if you were the top ranked pirate?

Write a computer program to input two integers, G and P , and to output P integers, indicating the division of gold that you would suggest for the P pirates (The pirates are ordered highest to lowest, so as the highest ranked pirate, the number of gold coins you get should be the first integer)

Examples

(1) $G=5$ $P=2$

Answer = 5 0

You should suggest that you get all the gold. Since you would vote for your own plan, that counts as "at least half the crew", therefore it will be accepted.

(2) $G=10$ $P=3$

Answer = 9 0 1

You should suggest that you get 9 gold coins and the bottom ranked pirate gets 1. Let's number the pirates 1 to 3, 1=highest ranked, 3=lowest ranked. If your plan fails, then pirate 2 will take all the gold, so it is in the interest of pirate 3 to vote for your plan, since he gets 1 coin (otherwise he would get nothing).

You can assume the input will be two positive integers within these ranges:

$P \leq 1,000$

$G \leq 1,000,000$

$G > (0.5 \times P)$
