Thomas Jefferson
Invitational Open in Informatics

Sample Problems

Version 2.01.1

By programmers, For programmers

Preface

Welcome to the TJ IOI Sample Problems document. Here, you will find a few problems of both the theoretical and practical type. Not all categories will be covered (see the Study Materials for other categories), but we hope that these problems will give you a better idea of what to expect at the competition. Enjoy!

Feel free to contact us (`tjioiofficers@gmail.com`) if you have questions or concerns, or if you would like some more sample problems. In addition, we greatly appreciate any feedback about the difficulty level, problem types, or anything.

Programming Problems

The practical problems require programs as solutions. The format of the problems in this document will be similar to the ones on the competition. All programs will be tested on 5–10 test cases. These test cases are generally more difficult than the example cases and may be tricky (but legal), so be sure to consider cases that might mess up your programs. There are many solutions for each problem, but we have provided just one written in Java (more languages available on request). The general guideline is that your program should read the input from standard input and output to standard output, as if one were typing in the input on a console. Be sure to follow the exact input and output format specifications. During the competition, all programs will have 30 seconds to run.

Theoretical Problems

These contain several problems with a short answer format. We may decide to include additional problems without backstory, or we may decide to use similarly formatted problems during the competition. However, these problems should give a better idea of the test in terms of problem types, difficulty, and the thinking required to solve the problems.

Solutions

Sample solutions are available in the solutions document.

Contents

Sample Programming Problem 1

Problem Statement: Candy Piles

William starts with three piles of candy. The first pile contains A pieces of candy, the second pile contains B pieces of candy, and the third pile contains C pieces of candy. William then buys enough candy to create the fourth pile, which contains an amount of candy equal to the first, second, and third piles combined. The fifth pile contains as much as the second, third, and fourth piles combined. William continues this pattern, with each new pile containing as many pieces of candy as the previous three piles. How many pieces of candy are in the N^{th} pile?

Input and Output Format

Input:

- Line 1: Four space-separated integers: A, B, C, and N. $0 \leq A, B, C \leq 10$, $1 \leq N \leq 30$.

Output:

- Line 1: One integer that is the number of pieces of candy in the N^{th} pile.

Examples

Input

```
1 2 3 5
```

Output

```
11
```

The fourth pile holds $1 + 2 + 3 = 6$ pieces of candy, and the fifth pile holds $2 + 3 + 6 = 11$ pieces of candy. The answer being sought is the size of the fifth pile, 11.

Input

```
5 5 5 10
```

Output

```
525
```

Sample Programming Problem 2

Problem Statement: Choosing Cows

Farmer John wants to choose 4 cows to represent his farm at the TJ IOI. The programming level of each cow represents its ability to do well in the competition and is represented by an integer from 1 to $10,000$. Farmer John has an infinite number of cows of each programming level. However, he does not want too high of an intelligence concentration on his team or else the team will be unfair. Thus, he wants to make sure that the sum of the squares of the intelligence levels of the four cows is less than or equal to N. How many ways are there for Farmer John to choose his team? Teams are ordered and two teams differ if the programming level of a cow in one position differs from the programming level of the cow from the other team in the same position. $\{1,4,1,2\}$ differs from $\{1,2,1,4\}$, for example.

Input and Output Format

Input:

- Line 1: One integer: N. $4 \leq N \leq 1,000,000$.

Output:

- Line 1: One integer that is the number of ways to choose the team of cows. Watch out—this number might be too large to fit in a 32-bit integer.

Examples

Input

```
7
```

Output

```
5
```

The five possible teams are $\{1,1,1,1\}$, $\{1,1,1,2\}$, $\{1,1,2,1\}$, $\{1,2,1,1\}$, and $\{2,1,1,1\}$.

Input

```
50
```

Output

```
467
```

Sample Programming Problem 3

Problem Statement: Grassy Patches

James wants to plant trees on the field! However, not all patches of grass are suitable for planting. On James's field, which is a grid containing N rows of N grass patches, a grassy patch is either "good" or "bad." A bad patch is marked with a 'B' and a good patch is marked with a 'G'. Two good patches are connected if it is possible to reach one good patch from the other by traveling only up, left, right, and down through other good patches. A "good cluster" is a group of connected patches that are all good. James can only plant trees in good clusters that contain at least 3 good patches. Help James count the number of good clusters that he can plant trees in.

Input and Output Format

Input:

- Line 1: One integer: N. $1 \leq N \leq 100$.
- Line 2...$N+1$: Line $i+1$ contains N characters, all either 'B' or 'G', that describe row i of the field.

Output:

- Line 1: One integer that is the number of good clusters on the field with at least 3 good patches.

Examples

Input

```
4
GGBB
GBBB
BBBG
GBGG
```

Output

```
2
```

The upper left corner and the bottom right corner are both part of good clusters of size 3. There is also a good cluster using the bottom left good patch, but that one is not large enough to hold a tree (it has a size of 1).

Input

```
7
BGGGBBG
GGBBBGB
BGBBGBB
BBBBGBB
BGGBBBG
BGGGGBG
GBBBBBG
```

Output

```
3
```

Sample Theoretical Problem 1

Three alien races—the binsarians (who use a base-2 number system), the octonians (who use base-8), and the hexons (who use base-16)—are meeting together and wish to know what the total population amongst the three races is. Unfortunately, the three alien races each counted their populations in their own base number systms, and they need your help to find the total population.

Race	**Population**
binsarians	1011_2
octonians	201_8
hexons	$5C_{16}$

1. What is the total population in base-10?

2. Each race would also like to know the total population in its own number system. Using the base-10 population calculated in part 1, what is the figure in base-2, base-8, and base-16?

3. There is actually a fourth alien race, the ternops. They count in base 3. The population of the ternops is equal to the total population of the binsarians and the octonians. Find the population of the ternops in base 3.

Sample Theoretical Problem 2

The nefarious wizard Sreenath has devised a set of problems to ensnare the valiant knight Alex. Sreenath's twisted mind has produced several convoluted recursive riddles. The riddles will sap Alex's resolve if he cannot solve them. Help Alex defeat his foe.

1. Sreenath demands from Alex the value of $f(20)$ where

$$f(x) = \begin{cases} f\left(\left\lfloor \frac{x}{2} \right\rfloor\right) & x = 39 \\ f(x+1) & 20 \leq x < 39 \\ 1 + f(x-1) & 0 < x < 20 \\ 0 & x \leq 0 \end{cases}$$

   where $\lfloor x \rfloor$ is the greatest integer less than x.

2. One of Sreenath's minions, a Srnth, is standing 7 units away from Alex. Srnths can hop across distances of 1 or 2 units. If the Srnth moves always straight toward Alex, how many sequences of 1-hops and 2-hops can it take to reach Alex?

Sample Theoretical Problem 3

Albert, Alec, and Alex are engineers. In particular, they are electrical engineers. Recently, they have been hired by Farmer Arjun to fix the lighting system in the farmer's house. Farmer Arjun has not been having a great year so far, so he has only one light bulb, which is controlled by switches conveniently labeled "A" through "F". The switches form a network that turns the light bulb on if the networks evalutes to true, and off if false. Help Albert, Alec, and Alex find which states turn the light bulb on, so that they can avoid states that result in electrocution.

For each of the following problems, assume that the boolean algebra expression given reflects the network of switches. Please leave your answers as ordered n-tuples. You may use $*$ to represent a value that can be either 0 or 1.

1. $A + B$
2. ABC
3. $\overline{A \oplus B}$
4. $A(B + AC + \overline{A})$
5. $\overline{D}E + AB + \overline{D} + 0 + ABC$
6. $((A + B)A) + \overline{(BA) + B}$

Sample Theoretical Problem 4

Alex is competing in the Thomas Jefferson Invitational Open in Informatics and is on the practical section of the contest. He wants to make sure that his programs run within the 30-second runtime limit. For each of the code segments below, determine the runtime efficiency in big-O notation. Assume that all necessary header files have been included, all variables have been declared, and the program compiles with no error.

1.
```
for(int i=0;i<N;i++){
  int a=0,b=1;
  while(a+b<i){
    a+=b;
  }
}
```

2.
```
int f(int n){
  if(n==0){
    return 1;
  }
  return f(n-1)+f(n-1);
}
int main(){
  ⋮ // has efficiency of O(1)
  f(N);
}
```

3. For $1 \le M \le 10000000$ and $1 \le N \le 25000000$
```
for(i=0;i<M;i++){
  a*=a;
}
for(i=0;i<N;i++){
  b+=b;
}
```

4.
```
for(i=0;i<V;i++){
  for(j=0;j<V;j++){
    for(k=0;k<V;k++){
      path[j][k]=min(path[j][k],path[j][i]+path[i][k]);
    }
  }
}
```

Sample Theoretical Problem 5

Alex the Aardvark and Arjun the Armadillo are questing for treasure when they meet Saketh the Sponge at the shoreline. Being friends, Saketh agrees to help Alex and Arjun cross the great Guralese Ocean. However, because Saketh likes puzzles (and difficult math problems) very much, he has placed several buoys joined by some logs to help Alex and Arjun cross the ocean. Of course, Alex and Arjun can't swim, so they must walk on the logs and buoys.

Given each of Saketh's descriptions of the layout of the logs and buoys, help Alex and Arjun draw out a map. Note that Alex and Arjun start on the beach and must reach the island. Note that there will always be at least one buoy connected to the beach and one to the island, which do not require the use of logs.

1. Saketh says: "I've placed 4 buoys and 4 logs. buoy A is right next to the beach, and is connected to buoy B and no other buoy. buoy C and D are both connected to buoy B, and buoy D is next to the island. Furthermore, C and D are each connected to another buoy, which is different between them."

2. Seeing that Alex and Arjun quickly figured out the map, Saketh decided to play even more games, telling them, "Sorry, guys, I was kidding. There are way more buoys and logs - 10 buoys and 15 logs total, to be exact. In fact, the 16-buoy, 15-log structure resembles two complete binary trees - one rooted near the beach and one rooted near the island. Also, only the 'bottomost' buoys in both trees are connected by a single log. Furthermore, both trees are balanced and complete, with the tree rooted near the beach having buoys A-H and the one near the island having buoys I-P."