Competitive Programming
Science Honors Program (SHP)
Session 3

Christian Yongwhan Lim

Saturday, October 17, 2026

Overview

- **Complete Search**

Complete Search

- Try them all!

Complete Search

- Try them all!

- Also known as **bruteforce**.

Palindromic Squares (USACO, Rob Kolstad)

- Palindromes are numbers that read the same forwards as backwards. The number 12321 is a typical palindrome.

- Given a number base B ($2 \leq B \leq 20$, in base 10), print all the integers N ($1 \leq N \leq 300$, in base 10) such that the square of N is palindromic when expressed in base B; also print the value of that palindromic square. Use the letters 'A', 'B', and so on to represent the digits 10, 11, and so on.

- Print both the number and its square in base B.

I/O Format

- **Input Format**: A single line with B, the base (specified in base 10).

- **Output Format**: Lines with two integers represented in base B. The first integer is the number whose square is palindromic; the second integer is the square itself. NOTE WELL THAT BOTH INTEGERS ARE IN BASE B!

Sample I/O

- **Input**
  - 10

- **Output**
  - 1 1
  - 2 4
  - 3 9
  - 11 121
  - ...

Discuss for few minutes!

Solution Idea?

- Try them all!

Sample Implementation

Chessboard and Queens (CSES 1624): Statement

Time limit: 1.00 s **Memory limit:** 512 MB

Your task is to place eight queens on a chessboard so that no two queens are attacking each other. As an additional challenge, each square is either free or reserved, and you can only place queens on the free squares. However, the reserved squares do not prevent queens from attacking each other.

How many possible ways are there to place the queens?

Input

The input has eight lines, and each of them has eight characters. Each square is either free (`.`) or reserved (`*`).

Output

Print one integer: the number of ways you can place the queens.

Chessboard and Queens (CSES 1624): Example

Example

Input:

```
........
........
..*.....
........
........
.....**.
...*....
........
```

Output:

```
65
```

Discuss for few minutes!

Solution Idea?

- Try them all!

Sample Implementation

```
#include <bits/stdc++.h>
using namespace std;

#include <ext/pb_ds/assoc_container.hpp>
using namespace __gnu_pbds;
template <class T>
using Tree = tree<T, null_type, less<T>,
rb_tree_tag, tree_order_statistics_node_update>;

typedef long long ll;
```

Sample Implementation (con't)

```
bool ok(vector<int> &p, vector<string> g) {
  for (int i=0; i<8; i++) {
    if(g[i][p[i]]=='*') return false;
    g[i][p[i]]='x';
  }
```

Sample Implementation (con't)

```
    for (int i=0; i<8; i++) for (int j=0; j<8; j++) {
      if(g[i][j]!='x') continue;
      for (int k=0; k<4; k++) {
        int ni=i, nj=j;
        for (int l=1; ; l++) {
          ni+=dx[k], nj+=dy[k];
          if(ni<0||ni>=8||nj<0||nj>=8) break;
          if(g[ni][nj]=='x') return false;
        }
      }
    }
    return true;
  }
```

Sample Implementation (con't)

```
int main() {
  ios::sync_with_stdio(0); cin.tie(0);cout.tie(0);
  vector<string> g(8);
  for (int i=0; i<8; i++) cin>>g[i];
  vector<int> p(8);
  for (int i=0; i<8; i++) p[i]=i;
  int ret=0;
  do { if(ok(p,g)) ret++; }
  while(next_permutation(p.begin(),p.end()));
  cout<<ret<<"\n";
  return 0;
}
```

Maximum Multiple Sum (1985B): Problem Statement

Given an integer n, find an integer x such that:

- $2 \leq x \leq n$.
- The sum of multiples of x that are less than or equal to n is maximized. Formally, $x + 2x + 3x + \cdots + kx$ where $kx \leq n$ is maximized over all possible values of x.

Input/Output Specs

Input

The first line contains t ($1 \le t \le 100$) — the number of test cases.

Each test case contains a single integer n ($2 \le n \le 100$).

Output

For each test case, output an integer, the optimal value of x. It can be shown there is only one unique answer.

Solution Idea

- To maximize the number of multiples of x less than n, it optimal to choose a small x, in this case, 2.
- The only exception is n=3, where it is optimal to choose 3 instead, since both 2 and 3 have only one multiple less than 3.

Sample Implementation

```
#include <bits/stdc++.h>
using namespace std;

int main(){
  int t; cin >> t;
  while(t--){
    int n; cin >> n;
    cout << (n == 3 ? 3 : 2) << endl;
  }
  return 0;
}
```


BREAK

Group Exercise

- You may work in groups with those sitting near you.
- Each person must submit their own solutions.
- You have **75 minutes**.
- Afterward, you will present your solution!


THANK YOU