

Задача В3. ТРИЪГЪЛНИЦИ

В равнината са дадени N триъгълника ($0 < N < 10\,000$) с целочислени координати на върховете си. Напишете програма **TR**, която извежда на стандартния изход броя на нееднаквите триъгълници. Входните данни се четат от стандартния вход, като на първия ред е дадена стойността на N . Следват N реда, всеки съдържащ по три двойки цели числа, разделени с интервал. Всяка двойка задава координатите на абсцисата и на ординатата на връх от поредния триъгълник. Координатите не надминават по абсолютна стойност 1000.

ПРИМЕР

Вход

```
3
0 0 1 0 0 2
0 0 2 0 0 1
0 0 1 0 0 1
```

Изход

```
2
```

Решение:

Два триъгълника са еднакви тогава и само тогава, когато имат съответно равни страни. В програмата последователно се прочитат координатите на върховете на поредния триъгълник от входния поток и се пресмятат квадратите от дължините на страните му. Тези стойности се нареждат в намаляващ ред и се извършва проверка, дали тази тройка числа е вече записана в използваната структура данни в програмата. Ако не е записана, тройката се добавя. Накрая се извежда броят на добавените тройки.

```
#include<iostream>
using namespace std;

const int N=10001;
int n;
struct tr{int a, b, c;};
tr d[N];

int main()
{ cin >> n;
  int k=0;
  for(int i=1; i<=n; i++)
  {
    int ax,ay,bx,by,cx,cy;
    cin >> ax >> ay >> bx >> by >> cx >> cy;
    tr p;
    p.a=(cx-bx)*(cx-bx)+(cy-by)*(cy-by);
    p.b=(cx-ax)*(cx-ax)+(cy-ay)*(cy-ay);
    p.c=(ax-bx)*(ax-bx)+(ay-by)*(ay-by);
    if(p.a<p.b) swap(p.a,p.b);
    if(p.a<p.c) swap(p.a,p.c);
    if(p.b<p.c) swap(p.b,p.c);
    bool f=true;
    for(int j=1; j<=k; j++)
      if((d[j].a==p.a)&&(d[j].b==p.b)&&(d[j].c==p.c))
        {f=false;break;}
    if(f){k++;d[k]=p;}
  }
  cout << k << endl;
}
```