

НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА

Областен кръг, 11 март 2007 г.

Тема за група В (10–11 клас)

Задача В1. РАЗСТОЯНИЕ

При зададени цели положителни числа M и N разглеждаме точките с целочислени координати (x, y) в равнината, за които $1 \leq x \leq M$ и $1 \leq y \leq N$. Образуваме всички отсечки, които имат краища в две различни точки от разглежданите.

Напишете програма **dist**, която за дадено дробно число d намира дължината на отсечка от описания вид, която е най-близка по стойност до стойността на d . Ако има повече от една такава стойност, да се изведе най-малката.

На един ред на стандартния вход са записани стойностите M , N и d , разделени с интервали ($1 < M < 3000$, $1 < N < 3000$, $0 < d < 10000$). Стойността на d е дадена като число с фиксирана десетична точка с най-много 4 цифри в дробната част.

Програмата трябва да изведе на стандартния изход търсената стойност като число с фиксирана десетична точка с точно 4 цифри в дробната част, получени чрез закръгляне с отсичане.

ПРИМЕР

Вход

4 3 3.2

Изход

3.1622

РЕШЕНИЕ

Лесно се съобразява, че съвкупността от всички възможни дължини на разглежданите отсечки се запазва същата, ако се ограничим само с отсечките, които имат начало, съвпадащо с началото на координатната система. Маркираме елементите на масив $b[k]$ за индексите k , чиято стойност се среща като квадрат на дължина на разглеждана отсечка. След това търсим най-малката по абсолютна стойност разлика между квадратен корен от k и d , където k пробягва всички стойности, за които $b[k]$ е маркиран. Извеждаме $\text{sqrt}(k)$ за такова k , за което гореописания минимум се достига за първи път. За точно спазване изискванията за форматиране на изхода според условието на задачата, програмата използва специално написана функция `print`.

```
#include<cstdio>
#include<cmath>
const int M = 3001, N = 3001; bool b[M*M+N*N];
int m,n; double d;

void print(double res)
{ char ss[100]; sprintf(ss, "%0.08lf", res);
  int i=0; int k=0; bool c=false;
  while(k<5)
  {printf("%c",ss[i]); if(ss[i]=='.') c=true;
   if(c) k++;
   i++;}
  printf("\n");}
```

```

}

int main()
{
    scanf("%d %d %lf", &m, &n, &d);
    for(int x=0;x<m;x++)
    for(int y=0;y<n;y++)
        b[x*x+y*y]=1;

    double e=1e10;
    int L=(m-1)*(m-1)+(n-1)*(n-1);
    int k0=0;
    for(int k=1;k<=L;k++)
        if(b[k])
            {double v=fabs(d-sqrt((double)k));
             if(v<e) {e=v; k0=k;}}
        }

    double res=sqrt((double)k0);
    print(res);
}

```

Емил Келеведжиев

Задача В2. ОКРЪЖНОСТИ

В равнината са дадени N окръжности ($0 < N < 10000$). Никои две окръжности нито се пресичат, нито се допират, но някои могат да съдържат вътре в себе си други окръжности. Напишете програма **circ**, която извежда дължината на най-дългата последователност от вложени една в друга окръжности.

На първия ред на стандартния вход е записано числото N . Следват N реда, всеки съдържащ по три цели числа, разделени с интервали. Тези числа задават съответно координатите (абсциса и ордината) на центъра и радиуса на поредната окръжност. Координатите и радиусите са в диапазона от -1000 до 1000 .

ПРИМЕР

Вход

```

5
1 1 1
20 20 13
15 15 5
15 15 2
20 20 1

```

Изход

```

3

```

РЕШЕНИЕ

Входните данни се прочитат в масив с елементи $c[i]$, съответстващи на дадените окръжности чрез дефинираната в програмата структура **cir**. Функцията `create_graph()` намира за всяка окръжност $c[i]$ номерата j на окръжностите, които се съдържат в нея, и ги записва последователно във вектора $c[i].next$. Едновременно се образува и обратната