// ConsoleApplication1.cpp : 定义控制台应用程序的入口点。

//

#include "stdafx.h"

#include<iostream>

#include <vector>

#define maxi 40000

using namespace std;

//int velement[maxi];

class Solution {

public:

bool canPlaceFlowers(vector<int>& flowerbed, int n) {

int len = flowerbed.size();

int count = 0;

for (int i = 0; i < len;)

{

while (flowerbed[i] == 0)

{

i++;

if ((i == len - 1 || i == 1) && flowerbed[i] == 0) // first two 0 or last two 0

count++;

else if (i < len - 1 && flowerbed[i] == 0 && flowerbed[++i] == 0) // three continuous 0

count++;

}

i++;

if (n <= count)

return true;

}

return false;

}

};

int main()

{

vector<int> v;

vector<int>::iterator it;

int velement;

int i = 0;

//char c;

int n;

Solution\* s = new Solution();

cout << "please intput the flowerbed: " << endl;

/\*while (scanf("%d", &velement[i++]) != EOF)

{

c = cin.get();

v.push\_back(velement[i]);

cout << "velement: " << velement[i] << endl;

cout << "vector: " << v[i] << endl;

if (c != ' ')

break;

}\*/

while (cin >> velement && i++ < 20000)

{

v.push\_back(velement);

if (cin.get() == '\n')

break;

}

cout << "the element in vector: " << endl;

for (it = v.begin(); it < v.end(); it++)

{

cout << \*it << ' ';

}

cout << endl<<"please intput n: " << endl;

cin >> n;

bool result = s->canPlaceFlowers(v, n);

cout << result << endl;

delete s;

}