ЗАДАНИЯ

 для проведения первого этапа республиканской олимпиады

 по учебному предмету «Информатика»

VI – XI классы

**Задача №1 «Конфеты»**

Однажды Петя, решая задачки, проголодался и решил пойти попить чай с конфетами. У него дома было много разных сортов конфет. Он выложил их всех ряд. Он решил их есть не по порядку, а особым образом.

Сначала он находит в ряду две рядом лежащие конфеты одного сорта, съедает их, потом сдвигает все конфеты и опять ищет две рядом лежащие одного сорта, съедает их, и т.д.

У Пети много конфет, он хочет узнать сможет ли он съесть их все, таким способом.

**Формат входного файла input.txt**

В первой строке входного файла находится целое число N (1 ≤ N ≤ 15000). Во второй строке находятся N целых чисел, разделенные пробелом, каждое означает сорт конфеты. Числа не превышают 1000 и больше 0.

**Формат выходного файла output.txt**

В выходной файл вывести ‘yes’ (без кавычек) если Петя сможет съесть все конфеты, а если не сможет, то вывести ‘no’ (без кавычек).

### *Пример*

|  |  |
| --- | --- |
| **input.txt** | **output.txt** |
| 21 1 | yes |
| 31 1 1 | no |
| 42 3 3 2 | yes |

***Программа***

var

a : array[1..15000] of integer;

s : array[0..16000] of integer;

n,i,r : integer;

begin

assign(input,'input.txt');

reset(input);

assign(output,'output.txt');

rewrite(output);

read(n);

for i:=1 to n do read(a[i]);

r:=0;

s[0]:=-1;

for i:=1 to n do

 if s[r]<>a[i] then

 begin

 r:=r+1;

 s[r]:=a[i];

 end

else r:=r-1;

if r=0 then write('yes')

 else write('no');

close(input);

close(output);

end.

***Тесты***

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Исходные данные** | **Ответ** | **Баллы** |
| 1 | 21 1 | yes | 10 |
| 2 | 31 1 1 | no | 10 |
| 3 | 1255 | no | 10 |
| 4 | 42 3 3 2 | yes | 10 |
| 5 | 71 1 2 3 3 2 4 | no | 10 |
| 6 | 101 2 3 3 2 1 1 5 5 1 | yes | 10 |
| 7 | 61 2 3 2 3 1 | no | 10 |
| 8 | 111 5 6 7 4 5 5 4 7 9 1 | no | 10 |
| 9 | 1011 12 13 14 15 16 17 18 19 20 | no | 10 |
| 10 | 221 4 6 7 8 5 5 8 3 3 3 3 7 6 6 6 6 6 4 1 1 1 | yes | 10 |

**Задача № 2 «Мишень»**

Вводятся координаты точек, в которые попали пять пуль. Необходимо определить, сколько очков "выбил" спортсмен, если известно, что мишень представляет собой концентрические окружности, центр которых находится в начале координат, радиус "яблочка" (10 очков) равен 20, а радиус каждого последующего кольца (9, 8,.., 1) на 10 больше предыдущего. Все, что выходит за пределы наибольшего кольца – "молоко" (0 очков). При попадании в линию кольца количество очков засчитывается по внутренней области.

Написать программу, где по введенным значениям координат точек, выводится количество набранных очков.

Уравнение окружности x2+y2=R2, где (x,y) координаты точки, R – радис окружности.

***Пример***:

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
|  -10 10 30 0100 110-20 -20 0 40 | 36 |

***Программа***

 var s,x,y: integer;

 r:real;

 a: array[1..5,1..2] of real;

 begin

 for x:=1 to 5 do

 for y:=1 to 2 do

 readln(a[x,y]);

 s:=0;

 for x:=1 to 5 do

 begin

 r:=sqrt(sqr(a[x,1])+sqr(a[x,2]));

 if r<=20 then s:=s+10 else

 if r<=30 then s:=s+9 else

 if r<=40 then s:=s+8 else

 if r<=50 then s:=s+7 else

 if r<=60 then s:=s+6 else

 if r<=70 then s:=s+5 else

 if r<=80 then s:=s+4 else

 if r<=90 then s:=s+3 else

 if r<=100 then s:=s+2 else

 if r<=110 then s:=s+1;

 end;

 writeln(s);

 end.

***Тесты***

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Ввод** | **Вывод** | **Баллы** |
| 1 | 00000 | 00000 | 50 | 20 |
| 2 | 1200100-100-100 | 0120100-100100 | 0 | 20 |
| 3 | 0400600 | 30050070 | 35 | 20 |
| 4 | 0-9015-1100 | -8010400-120 | 14 | 20 |
| 5 | 0-900-1100 | -800-1000-120 | 10 | 20 |

**Задача №3 «Нефтепровод»**

Некая далекая страна имеет выгодное географическое положение для транзита нефти. В связи с этим министерством торговли было решено построить нефтепровод. Рабочим была выдана карта местности, которая представляет собой декартову плоскость. Все точки этой плоскости можно однозначно задать координатами x и y. Глава министерства поставил на этой карте своей рукой две точки с координатами (x1,y1) и (x2,y2). Соответственно первая точка (x1,y1) – это начало нефтепровода, а вторая точка (x2,y2) – конец нефтепровода.

К сожалению, нефтепровод напрямую построить невозможно, т.к. на складах есть только отрезки труб различной длины. Более того, из-за специфики местной технологии трубы могут быть проложены только в направлении с севера на юг или с востока на запад и соединяются, образуя или прямую, или угол 90 градусов. Министерство задумалось над тем, возможно ли из отрезков труб длиной L1, L2, ..., LK построить нефтепровод, который соединял бы точки, отмеченные министром торговли. Естественно, количество отрезков на складах не безгранично, поэтому известно, что количество труб длины L1 – C1, длины L2 – C2, …, длины LK – CK. Ваша задача при всех существующих ограничениях построить нефтепровод с использованием минимального количества отрезков труб, если это возможно.

***Входные данные.*** Данные читаются из файла с именем ***input.txt***. В первой строке файла находятся числа x1, y1, x2, y2, K, затем 2K чисел: L1, L2, …, LK, C1, C2, …, CK.

***Выходные данные.*** В выходной файл ***output.txt*** вывести единственное число – минимальное количество отрезков труб, использованных при строительстве. Если это невозможно, то вывести -1

 ***Ограничения.*** 1 ≤ K ≤ 4, 1 ≤ x1, y1, x2, y2, Li ≤ 1000, 1 ≤ Ci ≤ 10, 1 ≤ i ≤ K, все числа целые. Время на прохождение одного теста – 0.5 секунды.

***Примеры***

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
|  5 5 5 6 1 2 10 | -1 |
| 20 10 60 50 2 70 30 2 2 | 4 |

***Тесты***

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | ***Входные данные*** | ***Выходные данные*** | **Баллы** |
| 1 | 1 1 100 1 1 99 1 | 1 | 10 |
| 2 | 1 10 10 10 1 10 1 | -1 | 10 |
| 3 | 150 150 10 10 1 140 10 | 2 | 10 |
| 4 | 2 3 8 6 2 3 6 3 1 | 2 | 10 |
| 5 | 400 400 300 300 2 620 600 10 10 | 20 | 10 |
| 6 | 30 30 45 17 3 1 4 2 4 5 6 | 10 | 10 |
| 7 | 387 256 1 1 3 3 137 937 10 10 10 | 13 | 10 |
| 8 | 1 1 1000 1000 3 995 996 997 10 10 10 | 6 | 10 |
| 9 | 300 300 332 332 4 1 2 4 8 10 10 10 10 | 8 | 10 |
| 10 | 400 400 415 415 4 1 2 4 8 10 10 10 10 | 6 | 10 |

***Программа***

var

 x1, y1, x2, y2, k: longint;

 l, c: array [1..4] of longint;

 qx, qy: array [1..4] of longint;

 sumX, sumY: longint;

 total: longint;

 best: longint;

procedure goY(i: longint);

var

 j: longint;

begin

 if total >= best then

 exit;

 if i = k then begin

 if sumY mod l[k] = 0 then begin

 if Abs(sumY) div l[k] <= c[k] - qx[k] then begin

 qy[k] := Abs(sumY) div l[k];

 Inc(total, qy[k]);

 if total < best then

 best := total;

 Dec(total, qy[k]);

 end;

 end;

 end

 else begin

 for j := -(c[i] - qx[i]) to (c[i] - qx[i]) do begin

 Inc(sumY, j \* l[i]);

 qy[i] := Abs(j);

 Inc(total, qy[i]);

 goY(i + 1);

 Dec(total, qy[i]);

 Dec(sumY, j \* l[i]);

 end;

 end;

end;

procedure goX(i: longint);

var

 j: longint;

begin

 if total >= best then

 exit;

 if i = k then begin

 if sumX mod l[k] = 0 then begin

 if Abs(sumX) div l[k] <= c[k] then begin

 qx[k] := Abs(sumX) div l[k];

 Inc(total, qx[k]);

 sumY := abs(y2 - y1);

 goY(1);

 Dec(total, qx[k]);

 end;

 end;

 end

 else begin

 for j := -c[i] to c[i] do begin

 Inc(sumX, j \* l[i]);

 qx[i] := Abs(j);

 Inc(total, qx[i]);

 goX(i + 1);

 Dec(total, qx[i]);

 Dec(sumX, j \* l[i]);

 end;

 end;

end;

var

 i: longint;

begin

 assign(input, 'input.txt'); reset(input);

 assign(output, 'output.txt'); rewrite(output);

 read(x1, y1, x2, y2, k);

 for i := 1 to k do

 read(l[i]);

 for i := 1 to k do

 read(c[i]);

 sumX := abs(x2 - x1);

 best := MAXLONGINT;

 goX(1);

 if best <> MAXLONGINT then

 writeln(best)

 else

 writeln(-1);

 close(output); close(input);

end.

**Задача №4 «Почтовая служба»**

Почтовая служба страны со странным названием «Хи в квадрате» решила наконец-то поставить почтовые ящики во всех городах на всей необъятной территории. Но тут же встал вопрос, а как это сделать оптимально? Под оптимальностью руководство почтовой службы понимает минимизацию максимального числа жителей приходящегося в среднем на один почтовый ящик в городе.

Пусть в стране имеется N городов и B почтовых ящиков. Необходимо распределить почтовые ящики по городам в соответствии с критерием оптимальности, предложенным руководством.

Рассмотрим пример. Пусть имеется 4 города с населением 120, 2680, 3400 и 200 жителей соответственно. В почтовой службе есть 6 почтовых ящиков. Чтобы оптимально распределить ящики между городами необходимо в первый город отправить 1 ящик, во второй – 2, в третий – 2, в четвертый – 1. Тогда среднее число жителей, приходящихся на один почтовый ящик, в первом городе будет 120, во втором – 1340, в третьем – 1700, в четвертом – 200. Таким образом, максимальное из средних будет 1700. Среди всех возможных вариантов распределения ящиков такой вариант будет наилучшим (т.е. менее чем 1700 жителей на один ящик добиться невозможно).

Помогите руководству почтовой службы государства «Хи в квадрате» оптимально распределить почтовые ящики по городам и определить, какое максимальное среднее число жителей приходится на один почтовый ящик среди всех городов. Определение этого числа позволит доказать руководству компетентность Вас как специалиста.

***Входные данные:***

Во входном файле input.txt записаны два натуральных числа N (1 ≤ N ≤ 500000) и B (N ≤ B ≤ 2000000).

Далее следует N строк, каждая из которых содержит одно натуральное число ai (1 ≤ ai ≤ 5000000) – численность населения в городе i.

**Выходные данные:**

В выходной файл output.txt требуется вывести единственное число – максимальное среднее число жителей, приходящихся на один почтовый ящик среди всех городов при оптимальном распределении ящиков. Если число получается не целое, то оно должно быть округлено вверх до ближайшего целого числа. Строка единственная.

***Примеры:***

|  |  |
| --- | --- |
| **input.txt** | **output.txt** |
| 2 7200000500000 | 100000 |
| 4 612026803400200 | 1700 |

***Программа***

var

 i,n,b,j,max,m:longint;

 A,C,T: array [1..500000] of longint;

begin

assign(input,'input.txt');reset(input);

assign(output,'output.txt');rewrite(output);

readln(n,b);

for i:=1 to n do begin

 read (A[i]); T[i]:=1; C[i]:=A[i];

 end;

for i:=1 to b-n do

begin

 max:=C[1]; m:=1;

 for j:=2 to n do if C[j]>max then begin max:=C[j]; m:=j; end;

 inc(T[m]);

 C[m]:=A[m] div T[m];

 if A[m] mod T[m]<>0 then inc(C[m]);

end;

max:=C[1];

 for j:=2 to n do if C[j]>max then max:=C[j];

write(max);

close(input);close(output);

end.

***Тесты***

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | ***Входные данные*** | ***Выходные данные*** | **Баллы** |
| 1 | 1 235 | 18 | 10 |
| 2 | 2 32579 | 40 | 10 |
| 3 | 3 1365646 | 10 | 10 |
| 4 | 4 1162929643 | 31 | 10 |
| 5 | 5 1692535493 | 19 | 10 |
| 6 | 10 38741436549476913071 | 14 | 10 |
| 7 | 15 763189283858885484461010592289 | 10 | 10 |
| 8 | 20 369982909910587323399339809158599216895712 | 47 | 10 |
| 9 | 25 10622594783824597233062365174674560934054255511465087 | 15 | 10 |
| 10 | 30 939738629626656131731271272124854219125896441526330114597966 | 18 | 10 |

**var**

 i,n,b,j,max,m:longint;

 A,C,T: array [1..500000] of longint;

begin

assign(input,'input.txt');reset(input);

assign(output,'output.txt');rewrite(output);

readln(n,b);

for i:=1 to n do begin

 read (A[i]);

 T[i]:=1;

 C[i]:=A[i];

 end;

for i:=1 to b-n do

begin

 max:=C[1]; m:=1;

 for j:=2 to n do if C[j]>max then begin max:=C[j]; m:=j; end;

 inc(T[m]);

 C[m]:=A[m] div T[m];

 if A[m] mod T[m]<>0 then inc(C[m]);

end;

max:=C[1];

 for j:=2 to n do if C[j]>max then max:=C[j];

write(max);

 close(input);close(output);

end.