

9-11

1.

2.

9-11

1. (100)

:

preliminary.

tests.

20 (5).

10

5

. (Turbo Pascal 7.0)

Program game ;

```

var
    n,k,x: longint;
    n1:real;
begin
    assign(input, 'game.in');
    reset(input);
    assign(output, 'game.out');
    rewrite(output);
    readln(n,k);
    x := (round(sqrt(1 + 8 * n)) - 1) div 2;
    if ((x + 1) * (x + 2) <= 2 * n) then x:=x-1;
    if (k <= x) then writeln(n - k * (k + 1) div 2)
    else
    begin
        n:=n - x * (x + 1) div 2;
        while (x + n * 2 < k) do
        begin
            x := x + 2 * n + 1;
            n := x;
        end;
        n := n - (k - x) div 2;
        if (k - x) mod 2 <> 0 then n:=n+k;
        writeln(n);
    end;
    close(input);
    close(output);
end.

```

2. (100)

:

.

$\min(A_1, B_1)$

$\min(A_1, B_3)$

$\min(B_1, A_3)$

$\min(A_2, B_3)$

$\min(B_2, A_3)$

$\min(A_3, B_3)$

preliminary.

tests.

50 (2).

().

2 , 50 .

. (Turbo Pascal 7.0)

Program Chips;

var a1, a2, a3, b1, b2, b3, ans : integer;

```

procedure pair(var a : integer; var b : integer);
var m : integer;
begin
    if a<b then m := a else m:=b;
    ans := ans + m;
    a := a - m;
    b := b - m;
end;

begin
    assign (input, 'chips.in');
    reset(input);
    assign(output, 'chips.out');
    rewrite(output);
    read(a1, a2, a3);
    read(b1, b2, b3);
    ans := 0;
    pair(a1, b1);
    pair(a2, b2);
    pair(a1, b3);
    pair(b1, a3);
    pair(a2, b3);
    pair(b2, a3);
    pair(a3, b3);
    writeln(ans);
    close (input);
    close (output);
end.

```

3. (100)

:

.

.

5

,

(N):
N K .

N K.

, N
N div K. ,
. - (N div K) + 1. - N

mod K. :

.

preliminary. ,

,

tests.

10 (5).

10 .

2 .

. (Turbo Pascal 7.0)

var

K,N : integer;

begin

assign(INPUT, 'book.in');

reset(INPUT);

assign(OUTPUT, 'book.out');

rewrite(OUTPUT);

read(K,N);

if N **mod** K = 0 **then** write(N **div** K, ' ', K)

else write (N div K + 1, ' ', N **mod** K);

close(INPUT);

close(OUTPUT);

end.

6

:

read(K,N) ;

write((n-1) div k) + 1, (n-1) mod k + 1) ;

4.

:

.

,

A.

0.

,

0.

,

,

,

.

1.

,

,

,

,

1.

,

,

.

2,

. . .

,

B.

,

,

.

(

).

.

,

(set).

l[i] -

,

i-

(

1 -

), current -

,

step

,

.

next

,

,

(step+1)

,

.

-

,

current

next.

-

,

-

step

```

    ,
    i,
    i (step+1) ,
    . , current*
l[i] ( ,
), l[i] next:
next:=next+l[i]; {
, M ,
A B (M-1)
,
(A B ).

```

. (Turbo Pascal 7.0)

var

```

N,M,A,B,i,j,k,t,step : integer;
next,emptyset, current : set of 1..100;
l : array [1..20] of set of 1..100;

```

begin

```

assign(INPUT,'buses.in');
reset(INPUT);
assign(OUTPUT,'buses.out');
rewrite(OUTPUT);
read(N,M);
for i:=1 to M do
begin
    read(k);
    for j:=1 to k do
    begin
        read(t);
        Include(l[i],t);
    end;
end;

```

```

end;
read(A,B);
emptyset:=[];
step:=0;
current:=[A];
for step:=0 to M-1 do
begin
    next:=current;
    for i:=1 to M do
        if current*l[i]<>emptyset
        then next:=next+l[i];
        if B in next then
            begin
                write(step); close(output);exit;
            end;
        current:=next;
    end;
write(-1);
close(INPUT);
close(OUTPUT);
end.

```


, . - ,

.

, -

, .

Core 2 Duo 2.4 .

, .

, .

Pentium 3 866

2 ,

, 4 . ,

Core i7 3

0,2 , , 1 .

4

« -

».

,

4

,

-

,

.

,

:

)

;

)

exe-

,

,

)

.

,

,

,

,

(

)

(

).

0

.

,

,

.

1)

,

.

,

,

:

--	--

Borland Delphi 7.0	dcc32 - < >
Free Pascal 2.4.0	fpc < >
Visual 2005	cl /O2 / < >
GNU 4.4.0 (MinGW)	gcc -O2 - -W1, -- stack=67108864 < >
Visual C++ 2005	cl /O2 /EHs /TP < >
GNU C++ 4.4.0 (MinGW)	g++ -O2 - ++ -W1, -- stack=67108864 < >
Visual Basic 2005	vbc < >
Borland Pascal 7.0	b < >
Borland 3.1	bc -ml < >
Borland C++ 3.1	bc -ml < >

2)

64

1

,

,

..

« »

,

.

.

,

,

,

.

«

»

.

,

-

,

« ».

,

.

.

,

.

.

,

,

-

.

,

,

-

.

.A

,

-

.

tester080311

TESTER build

080311 (!test.exe,
http://acm.timus.ru/tester).

, 2003-2008,

readme.txt.

4

«sweets»,

«7-8

».

:

1)

«preliminary»,

!test.exe,

«01», «02», . . .
 «01. », «2. », . . .
 2) «tests»,
 «01.in»,
 «02.in» . . . «01.out», «02.out»
 . . .

3) Turbo Pascal 7.0 (
 sweets.pas),

TESTER build 080311,
 Yandex.

TESTER build 080311

Yandex.

TESTER build 080311

«preliminary»

:

1) «preliminary»

;

2) !test.exe.

« »,

«01», «02», . .

«01.A», «02.A», . .

(game.exe, chips.exe, book.exe busses.exe).

« »

!test.ini

InputFile OutputFile.

– RunProgram.
 1 9-11 (,
):

RunProgram = game.exe

TestInput = 00.in

TestAnswer = 00.out

TestNumber = Auto

TimeLimit = 1.0

MemoryLimit = 64.0

Checker = #token

HideCheckerOutput = No

HideProgramOutput = Yes

GenAnswers = Auto

Interrupt = No

Pause = Yes

InputFile = game.in

OutputFile = game.out

; === Краткая помощь (более детально читайте файл readme.txt) ===

;RunProgram Исполнимый файл участника для тестирования

;TestInput Маска имени входного файла. Последние '00' или

; '000' будут заменяться на текущий номер теста

;TestAnswer Маска имени выходного файла. Mask of test answer file.

; '000' будут заменяться на текущий номер
 теста
 ;TestNumber Количество тестов или AUTO для
 автоматического
 ; определения этого количества
 ;TimeLimit Максимальное время в секундах для работы
 программы
 ;MemoryLimit Максимальный размер памяти,
 разрешенный программе
 ; или NO, чтобы убрать ограничение памяти
 ;Checker Программа проверки, используйте #TOKEN
 ;HideCheckerOutput Если значение этого параметра равно
 YES или 1,
 ; то вывод проверяющей программы
 подавляется, и на
 ; экран выводится только либо "ok" либо "wrong
 ; answer"
 ;HideProgramOutput Если значение этого параметра равно
 YES или 1, то
 ; консольный вывод проверяемой программы
 подавляется.
 ;GenAnswers Если значение этого параметра равно YES
 или 1, то
 ; содержимое текущих файлов с выходными
 данными
 ; будет заменено результатами работы
 программы на
 ; каждом тесте.

;Interrupt Если значение этого параметра равно YES или 1,
 то работа
 ; Тестера завершается после первого неудачного
 запуска.
 ; NO или 0 - продолжение тестирования в любом
 случае
 ; (рекомендуемое)
 ;Pause Если значение YES или 1, то по окончании работы
 Тестера
 ; происходит задержка экрана. NO или 0 - работа
 завершается
 ; немедленно
 ;InputFile Файл, из которого тестируемая программа будет
 читать
 ; входные данные (имя входного файла).
 ;OutputFile Файл, в который тестируемая программа будет
 выводить
 ; результаты работы (выходной файл).
 ;ScorePerTest Баллы, начисляемые за каждый пройденный
 тест.
 ; Необязательный параметр. Значение по умолчанию -
 0.
 ;ScoreAccepted Баллы, начисляемые за прохождение всех
 тестов.
 ; Необязательный параметр. Значение по умолчанию -
 ; ScorePerTest, умноженное на количество тестов. В
 случае

; отсутствия обоих параметров, баллы не подсчитываются.

3)

- 1) ;
- 2) , (,);
- 3) (, :
- 4) (,)

10. 1, 4 – 20, 2 – 50, 3 –
 «XY.in» tests.
 1 N, N —
 «XY.out». XY —
 «07.in», — «15.out»,
 «07.in»,
 — «15.out».

1. : 100 —
 , 1 5
 , 2 2 , 3
 10 , 4 5 .

2.

,

.

3.

.

,

100

,

,

400

.