

ЗИМНИ МАТЕМАТИЧЕСКИ ПРАЗНИЦИ – РУСЕ’ 2006

Йода говори език, наречен „галактически бейсик” и се отличава с това, че често разменя словореда в изреченията си. Например, вместо „Аз съм Йода, старши джедай”, той би казал „Джедай старши, Йода съм аз.”

Тъй като за много цивилизации неговият език остава неразбран, Йода отправя предизвикателство към юбилейното състезание „Зимни математически празници”. Напишете програма, която извежда думите от дадено изречение в обратен ред.

Йода е едно от най-мъдрите същества във Вселената, затова е изпратил и алгоритъм за решаването на задачата:

1. От клавиатурата ред да прочетеш от входни данни трябва.
 2. Символ по символ обърни ти после реда цял. Например „АЗ СЪМ ЙОДА” става „АДОЙ МЪС ЗА”.
 3. Във всяка дума поотделно после обръщай символите ти.
 4. И „ЙОДА СЪМ АЗ” накрая трябва да се появи.
- И нека Силата да бъде с теб!

В трансгалактическото съобщение от Йода, пише, че и авторът на най-добре написаната програма ще получи точки на състезанието и безплатно обучение за джедай.

Вход: От клавиатурата се въвежда едно изречение (не повече от 255 символа), съдържащо само главни букви от А до Z. Думите в изречението са разделени с **по** точно един интервал.

Изход: Програмата трябва да изведе изречението с разменени думи.

ПРИМЕР:

Вход:	Изход:
AZ SAM YODA	YODA SAM AZ

Решение:

Тъй като алгоритъмът е описан в задачата, той трябва просто да бъде реализиран.

```
var s,ss:string;
    i,len:byte;
    start,wlen:integer;

procedure swap(var a,b:char);
var c:char;
begin
  c:=a;
  a:=b;
  b:=c;
end;

begin
  readln(s);
  len:=length(s);
  { Reverse the whole string }
  for i:=(len div 2) downto 1 do
    swap(s[i],s[len-i+1]);
  { Reverse each word }
  start:=-1;
  repeat
    start:=start+wlen+1;
    wlen:=0;
```

ЗИМНИ МАТЕМАТИЧЕСКИ ПРАЗНИЦИ – РУСЕ’ 2006

```
while (s[start+wlen+1]<>' ') and (start+wlen<=len) do inc(wlen);
ss:=copy(s,start+1,wlen);
for i:=length(ss) div 2 downto 1 do
  swap(ss[i],ss[length(ss)-i+1]);
write(ss);
if start+wlen<=len then write(' ') else writeln;
until start+wlen>len;
end.
```

Задача D2. ПЕРДЕ

Марийка си купила ново перде за вкъщи. За да изглежда добре окачено, тя иска кукичките, на които се закачва пердето, да са на равни разстояния една от друга. В горния край на пердето има зашити халки на равни разстояния, за които могат да **се** закачват кукичките. Разбира се, за да не виси пердето от двата края, първата кукичка трябва да се закачи за първата халка, а последната кукичка - за последната халка. Нека номерираме халките с целите числа и най-лявата да получи номер едно.

Помогнете на Марийка, като напишете програма **PERDE**, която, по зададен брой на халките и брой на кукичките, определя номерата на тези халки, за които са закачени кукички.

Вход: От стандартния вход се въвеждат две цели положителни числа - **A** и **B**, където **A** е броя на халките, а **B** - броя на кукичките ($1 < A < 1000$, $1 < B < A$).

Изход: На стандартния изход се извежда на екрана нарастваща редица от номерата на тези халки, за които са закачени кукички.

Тъй като не винаги е възможно всички кукички да са закачени на халки, които се намират на равни разстояния една от друга, то е необходимо поне разликата между най-голямото и най-малкото разстояние между две съседни халки да бъде минимална.

ПРИМЕР:

Вход:	Изход:
10 4	1 4 7 10
11 4	1 5 8 11

Решение:

Задачата се свежда до намиране на частното и остатъка от делението на A на B.

```
#include <stdio.h>
```

```
int main(int argc, char **argv) {
  int n, k;
  scanf("%d %d", &n, &k);
  printf("n=%d k=%d\n", n, k);

  int m = n - k;
  int d = k - 1;
  int l = m % d;
  int s = m / d;
  printf("m=%d d=%d l=%d s=%d\n", m, d, l, s);

  int i = 0;
```