

Zadaci

Zadatak	MODULO	HERMAN	OKVIRI	SLIKAR	BOND	DEBUG
Izvorni kôd	modulo.bas modulo.pas modulo.c modulo.cpp	herman.bas herman.pas herman.c herman.cpp	okvir.bas okvir.pas okvir.c okvir.cpp	slikar.bas slikar.pas slikar.c slikar.cpp	bond.bas bond.pas bond.c bond.cpp	debug.bas debug.pas debug.c debug.cpp
Ulazna datoteka	standardni ulaz (tipkovnica)					
Izlazna datoteka	standardni izlaz (zaslon)					
Memorijsko ograničenje (heap)	32 MB	32 MB	32 MB	32 MB	32 MB	32 MB
Memorijsko ograničenje (stack)	8 MB	8 MB	8 MB	8 MB	8 MB	8 MB
Vremensko ograničenje (po test podatku)	1 sekunda	1 sekunda	1 sekunda	1 sekunda	1 sekunda	1 sekunda
Broj test podataka	10	10	10	10	10	15
Broj bodova (po test podatku)	1	2	3	5	7	8
Ukupno bodova	10	20	30	50	70	120
	300					

Napomena: Vremensko ograničenje je mjereno na računalu baziranom na dva procesora AMD Athlon MP 2600+ i operativnim sustavom Linux. Program u C-u i C++-u treba kompajlirati sa slijedećim opcijama: **-O2 -lm -static**, dok program u Pascalu sa **-So -O1 -XS**. U programskom jeziku QBasicu-u dozvoljeno vrijeme izvršavanja programa je dvostruko veće.

1. MODULO

Za neka dva prirodna broja A i B izraz A modulo B je jednak ostatku pri dijeljenju broja A s brojem B. Tako recimo brojevi 7, 14, 27 i 38 postaju 1, 2, 0, 2 kad se gledaju modulo 3.

Od 10 unesenih brojeva, odredite koliko je različitih kada se gledaju modulo 42.

Ulazni podaci

10 prirodnih brojeva manjih od 1000, svaki u svojem redu.

Izlazni podaci

U prvoj i jedinoj liniji potrebno je ispisati broj različitih brojeva modulo 42.

Test primjeri

tipkovnica	tipkovnica	tipkovnica
1	42	39
2	84	40
3	252	41
4	420	42
5	840	43
6	126	44
7	42	82
8	84	83
9	420	84
10	126	85
zaslon	zaslon	zaslon
10	1	6

Pojašnjenje test primjera:

U prvom test primjeru brojevi modulo 42 iznose 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, što je 10 različitih brojeva.

U drugom test primjeru svi brojevi modulo 42 iznose 0, što je jedan broj.

U trećem test primjeru brojevi modulo 42 iznose 39, 40, 41, 0, 1, 2, 40, 41, 0, 1, što je 6 različitih brojeva.

2. HERMAN

Dokaz da matematičari često imaju previše slobodnog vremena su brojne geometrije koje su izmišljene kroz godine. Jedna od zgodnijih je *taxicab* geometrija koju je još u 19. stoljeću proučavao njemački matematičar Hermann Minkowski. U *taxicab* geometriji udaljenost dvije točke definira se kao:

$$D(T1, T2) = |x1 - x2| + |y1 - y2|$$

Sve ostale definicije jednake su kao i u običnoj geometriji, pa tako i definicije kružnice i kruga koje glase:

Kružnica je skup točaka jednako udaljenih od neke točke (koju nazivamo središtem kružnice).

Krug je skup točaka ravnine omeđen kružnicom.

Zanima nas kako se razlikuju površine dvaju krugova polumjera R od kojih je jedan u običnoj (euklidskoj) geometriji, a drugi u *taxicab* geometriji. Na vas je pala strašna odgovornost proučavanja ovog problema.

Ulazni podaci

U prvom i jedinom retku nalazi se polumjer R , prirodni broj manji od ili jednak 10000.

Izlazni podaci

U prvi redak treba ispisati površinu kruga polumjera R u euklidskoj geometriji.

U drugi redak treba ispisati površinu kruga polumjera R u *taxicab* geometriji.

Napomena: Dopušteno odstupanje od točnog (preciznog) rješenja je ± 0.0001 .

Test primjeri

tipkovnica 1	tipkovnica 21	tipkovnica 42
zaslon 3.141593 2.000000	zaslon 1385.442360 882.000000	zaslon 5541.769441 3528.000000

3. OKVIRI

Tekst pisan metodom “Okvirići Petra Pana” je tekst u kojem je svaki znak uokviren s okvirićem dijamantnog oblika, s tim da se okvirići susjednih znakova isprepliću. Svako slovo potrebno je uokviriti na sljedeći način ('X' je slovo koje želimo uokviriti) :

```
..#..
.#.#.
#.X.#
.#.#.
..#..
```

Međutim, takvo uokvirivanje bi bilo vrlo monotono te ćemo zbog toga svako treće slovo uokviriti tzv. “Wendy okvirićem”. Wendy okvirić izgleda ovako:

```
..*..
.*.*.
*.X.*
.*.*.
..*..
```

Pri ispreplitanju Wendy okvirića i Okvirića Petra Pana, Wendy okvirić je ljepši i zbog toga je na vrhu. Za primjer ispreplitanja pogledajte test primjere.

Ulazni podaci

U prvom i jedinom retku ulaza nalazit će se riječ od najviše 15 znakova. Svi znakovi će biti velika slova engleske abecede.

Izlazni podaci

Riječ iz ulaza napisana metodom “Okvirići Petra Pana”.

3. OKVIRI

Test primjeri

tipkovnica

A

zaslon

..#..

.#.#.

#.A.#

.#.#.

..#..

tipkovnica

PAS

zaslon

..#...#...*..

.#.#.#.#.*.*.

#.P.#.A.*.S.*

.#.#.#.#.*.*.

..#...#...*..

tipkovnica

ABCD

zaslon

..#...#...*...#..

.#.#.#.#.*.*.#.#.

#.A.#.B.*.C.*.D.#

.#.#.#.#.*.*.#.#.

..#...#...*...#..

4. SLIKAR

Zli Kaktus car dočepao se Čarobne bačvice i poplavio Čudesnu šumu! Slikar Paleta i tri mala ježa sada se moraju što brže vratiti do Dabrove kolibe, gdje će se skloniti od bujice.

Karta Čudesne šume se sastoji od R redaka i S stupaca. Na karti su prazna polja označena znakom '.', polja pokrivena vodom znakom '*', a polja na kojima se nalazi stijena znakom 'X'. Također, dva su istaknuta polja, polje na kojem se nalazi Dabrova koliba (označeno znakom 'D') te polje na kojem se nalaze Slikar i tri mala ježa (označeno znakom 'S').

Slikar Paleta i tri mala ježa se u svakoj minuti mogu s polja na kojem se nalaze pomaknuti na 4 susjedna polja (gore, dolje, lijevo, desno). Svake minute se također i poplava proširi, i to tako da svako prazno polje koje je susjedno barem jednom već poplavljenom polju postaje poplavljeno. Ni voda, ni Slikar Paleta i tri mala ježa ne mogu proći kroz polje na kojem se nalazi stijena. Naravno, Slikar Paleta i tri mala ježa ne mogu proći kroz poplavljeno polje te voda ne može poplaviti Dabrovu kolibu.

Napišite program koji će za unesenu kartu Čudesne šume ispisati **najkraće** vrijeme potrebno da Slikar Paleta i tri mala ježa sigurno dođu do Dabrove kolibe.

Napomena: Slikar Paleta i tri mala ježa se ne smiju pomaknuti u polje koje će te minute biti poplavljeno.

Ulazni podaci

U prvom retku nalazit će se R i S, prirodni brojevi manji od ili jednaki 50.

U sljedećih R redaka nalazit će se po S znakova ('.', '*', 'X', 'D' i 'S'). U cijeloj karti nalazit će se točno jedan znak 'D' i točno jedan znak 'S'.

Izlazni podaci

U jedan red treba ispisati **najkraće** vrijeme potrebno da Slikar Paleta i tri mala ježa sigurno stignu do Dabrove kolibe. Ukoliko je to nemoguće potrebno je ispisati riječ "KAKTUS".

Test primjeri

tipkovnica 3 3 D.*S.	tipkovnica 3 3 D.*S	tipkovnica 3 6 D...*. .X.X..S.
zaslon 3	zaslon KAKTUS	zaslon 6

Pojašnjenje drugog test primjera: Najbolje što Slikar Paleta i ježevi mogu je pratiti donji, a zatim lijevi rub karte i biti poplavljeno minutu prije nego što stignu do Dabrove kolibe.

5. BOND

Svi znaju za tajnog agenta 007, popularnog Bonda (Jamesa Bonda). Manje poznata činjenica je da on zapravo nije obavio većinu svojih zadataka, već ih u njegovo ime obavljaju njegovi manje poznati rođaci, Jimmyji Bondovi. Bondu (Jamesu Bondu) je postalo zamorno svaki put kada dobije popis zadataka raspoređivati Jimmyje Bondove pa je uposlio vas da automatizirate taj postupak.

Bond (James Bond) početkom mjeseca dobije popis zadataka koje mora izvršiti. Uz pomoć detaljne dokumentacije s prošlih zadataka Bond (James Bond) također zna koliku vjerojatnost za uspješno izvršavanje nekog zadatka ima pojedini Jimmy Bond. Vaš program treba obraditi te podatke i svakom Jimmyju Bondu pridružiti neki zadatak tako da vjerojatnost da se **svi** zadaci **uspješno** izvrše bude **što je moguće veća**.

Napomena: Vjerojatnost da svi zadaci budu uspješno izvršeni jednaka je umnošku vjerojatnosti uspjeha pojedinih zadataka.

Ulazni podaci

U prvom retku unosi se prirodni broj N , broj Jimmyja Bondova i zadataka ($1 \leq N \leq 20$). Nakon toga slijedi N redaka po N cijelih brojeva između 0 i 100 (uključivo), gdje j -ti broj u i -tom retku predstavlja vjerojatnost da Jimmy Bond broj i uspješno izvrši zadatak j , izraženu u postocima.

Izlazni podaci

U prvi redak izlaza treba ispisati vjerojatnost uspješnog izvršavanja svih misija pri optimalnom rasporedu, izraženu kao postotak.

Napomena: Dopusšteno odstupanje od točnog (preciznog) rješenja je ± 0.000001 .

Test primjeri

tipkovnica 2 100 100 50 50 zaslon 50.000000	tipkovnica 2 0 50 50 0 zaslon 25.000000	tipkovnica 3 25 60 100 13 0 50 12 70 90 zaslon 9.100000
--	--	---

Pojašnjenje trećeg test primjera: Ukoliko Jimmyje Bondove rasporedimo tako da prvi dobije treći zadatak, drugi prvi zadatak, a treći drugi zadatak, ukupna vjerojatnost uspjeha je $1.0 * 0.13 * 0.7 = 0.091 = 9.1\%$. Svi ostali rasporedi imaju manju vjerojatnost uspješnog završavanja svih zadataka.

6. DEBUG

Prilikom debugiranja jednog programa Mirko je uočio da se greška u programu pojavljuje samo kada se u memoriji programa pojavi takozvani kvadratni ubojica. Memorija programa je matrica od R redaka i S stupaca u kojoj se nalaze samo nule i jedinice. Kvadratni ubojica je kvadratni podskup memorije **veći od jednog znaka**, koji izgleda jednako kada se okrene za 180 stupnjeva. U sljedećoj matrici postoje 3 kvadratne ubojice:

10101010	101...
11100101	...00.	111...
10100100.	101...
memorija	ubojica	ubojica	ubojica

Mirka zanima postoji li nekakva veza između dimenzije najvećeg kvadratnog ubojice i greške u njegovom programu. Napišite program koji će pomoći Mirku i za zadanu memoriju ispisati dimenziju **najvećeg** kvadratnog ubojice. Dimenzija kvadratnog ubojice je broj redaka (koji je jednak broju stupaca) od kojih se sastoji. U gornjem primjeru dimenzije su redom 2, 2, 3.

Ulazni podaci

U prvom retku ulaza nalazit će se R i S, prirodni brojevi manji od ili jednaki 300.
U sljedećih R redaka nalazit će se po S znakova, '0' ili '1' (bez razmaka).

Izlazni podaci

U prvi i jedini redak treba ispisati dimenziju najvećeg kvadratnog ubojice. Ukoliko nema kvadratnih ubojica, ispišite -1.

Test primjeri

tipkovnica 3 6 101010 111001 101001 zaslon 3	tipkovnica 4 5 10010 01010 10101 01001 zaslon 3	tipkovnica 3 3 101 111 100 zaslon -1
--	--	--