

00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
00000000
10101010

Uitvoer bij de voorbeeldinvoer

10101011

Voorronde Programmeren 1994

Probleem B - Kinderrijmpjes

Neem een groep kinderen. Om een kind uit de groep te selecteren voor een spelletje wordt een kinderrijmpje gebruikt. De selectie vindt als volgt plaats. Een kind begint het rijmpje te zingen met de eerste lettergreep van de eerste regel. Zijn/haar buurman/vrouw in de richting van de klok zingt de volgende lettergreep van die regel. Het kind dat de laatste lettergreep van een regel zingt, stapt uit de groep en wordt niet geselecteerd. De groep (zittend in een cirkel uiteraard) wordt zo kleiner. De buurman/vrouw van de uitgevallene (weer met de klok mee) gaat verder met de eerste lettergreep van de volgende regel (of weer met de eerste regel als we waren aangekomen bij het eind van het rijmpje). Het zingen gaat verder totdat er 1 kind over is; dit is het geselecteerde kind.

Laat N , $N > 0$, het aantal kinderen zijn, deze kinderen zijn genummerd van 0 tot en zonder N , met de richting van de klok. Het kinderrijmpje bevat k , $k > 0$, regels, regel i , $0 \leq i < k$, bevat S_i lettergrepen ($S_i > 0$). Laat b , $0 \leq b < N$, het nummer zijn van het kind dat begint te zingen.

Het volgende voorbeeld is het selectie proces in het geval dat $N=9$, $k=2$, $S_0=3$, $S_1=4$ en $b=0$. (Bijvoorbeeld het rijmpje: Jij bent WEG, Dat is nu PECH.) Voor elke regel die gezongen is hebben we de nummers van de kinderen die nog in cirkel zitten gegeven, beginnend met het kind dat die regel begonnen is met zingen. Het kind dat uit de groep stapt is tussen haakjes gezet. Aan het eind van het proces is kind 3 geselecteerd.

```
0 1 (2) 3 4 5 6 7 8
3 4 5 (6) 7 8 0 1
7 8 (0) 1 3 4 5
1 3 4 (5) 7 8
7 8 (1) 3 4
3 4 7 (8)
3 4 (7)
3 (4)
```

Beschrijving van de invoer

Uw programma berekent het geselecteerde kind voor een serie gevallen, elk geval is gegeven door N , k , b , S_i . De volgende limieten gelden: $1 \leq N \leq 10.000$, $1 \leq k \leq 20$ en $1 \leq S_i \leq 10$ voor $0 \leq i < k$.

Dus in de invoer staat eerst het aantal keren dat een kind geselecteerd moet worden. Voor elke keer staan er daarna twee regels met op de eerste regel het aantal kinderen, het nummer van het kind dat begint te zingen en het aantal regels van het rijmpje. Op de tweede regel staan het aantal lettergrepen van elke regel van het rijmpje.

Beschrijving van de uitvoer

Elke regel bevat het nummer van het geselecteerde kind van het corresponderende geval in de invoer.

Voorbeeldinvoer

1
9 0 2
3 4

Uitvoer bij de voorbeeldinvoer

3

Voorronde Programmeren 1994

Probleem C - Doolhof

In het spel Snipes kunnen verschillende spelers binnen 1 veld tegen elkaar spelen. Doel van het spel is om zowel de snipe-nesten (generators), de snipes zelf en de andere spelers uit het speelveld te schieten.

Het speelveld bestaat uit een circulair veld van x by y vakjes. Circulair betekent hier dat de "randen" van het vierkant verbonden zijn met de "overkant". Als men zich op de bovenrand, bijvoorbeeld op positie $(3,1)$ bevindt, bevindt men zich na 1 stap noordwaarts (naar boven) op de onderrand op de positie $(3,y)$.

Aan het begin van het spel wordt een doolhof gecreerd. Een doolhof ontstaat door op de scheidslijn tussen twee vakjes a en b een muur aan te brengen, waardoor het onmogelijk is geworden om in 1 stap van a naar b te gaan. Het doolhof wordt zo gecreerd dat er geen eilanden ontstaan, dus ieder vakje is (op den duur) bereikbaar.

Opgave

Gevraagd wordt om gegeven de afmetingen van een doolhof, het maximale aantal muren te berekenen dat men kan plaatsen, waarbij ieder vakje nog bereikbaar blijft vanuit elk willekeurig ander vakje.

Beschrijving van de invoer

De invoer bestaat uit het getal n , gevolgd door n regels. Op iedere regel staat een probleem, bestaande uit de gehele getallen x en y , gescheiden door 1 spatie. Hierbij is het aantal kolommen gegeven door x ($1 \leq x \leq 1000$) en het aantal rijen door y ($1 \leq y \leq 1000$).

Beschrijving van de uitvoer

De uitvoer bestaat uit n regels met op iedere regel het maximaal aantal muren dat kan worden geplaatst bij het corresponderende probleem uit de invoer. De oplossing bestaat uit een getal, gepositioneerd op de eerste 8 posities, rechts uitgelijnd.

Voorbeeldinvoer

```
3
2 2
3 1
3 5
```

Uitvoer bij de voorbeeldinvoer

```
5
4
16
```

Voorrunde Programmeren 1994

Probleem D - Woordzoekers bouwen

Veel mensen verdoen hun tijd met het invullen van dingen als kruiswoord raadsels, doorlopers en andere spelletjes waarbij je woorden op, door en naast elkaar moet leggen. Wij informatici hebben natuurlijk geen tijd om zo'n puzzel echt met de hand op te lossen, wij schrijven daar een programma voor!

Opgave

Gegeven een rechthoek $x*y$ en een lijst met woorden. Is het mogelijk om alle woorden in de rechthoek te plaatsen. Plaats de woorden in de binnengekomen volgorde en probeer op de volgende manieren:

- Eerst horizontaal van links naar rechts
- Dan diagonaal omlaag van links naar rechts
- Dan verticaal omlaag
- Dan diagonaal omlaag van rechts naar links
- Dan horizontaal van rechts naar links
- Dan diagonaal omhoog van rechts naar links
- Dan verticaal omhoog
- Dan diagonaal omhoog van links naar rechts

Begin bij elk woord en elke manier steeds in de linkerbovenhoek. En werk regel voor regel steeds van links naar rechts de manier af. De eerste keer dat een woord kan worden geplaatst moet het ook meteen geplaatst worden. (Het eerste woord zal dus altijd zijn eerste letter in de linkerbovenhoek hebben, als het in de rechthoek past.)

Beschrijving van de invoer

De eerste regel van de invoer bestaat uit het aantal problemen. Voor elk probleem komt er dan een regel met de grote van de rechthoek x, y en het aantal woorden i . Daarna i regels met woorden die geplaatst moeten worden. Met $1 \leq x, y, i \leq 100$.

Beschrijving van de uitvoer

De uitvoer bestaat uit de x bij y rechthoek met daarin de woorden geplaatst zoals hierboven beschreven. Als je alle woorden hebt gehad, vul dan de nog lege plekken in met een X. Als een woord niet in de rechthoek past, geef dan in de uitvoer slechts de letters KN van "kan niet".

Voorbeeldinvoer

```
2
6 5 6
appel
kleden
kom
popen
kwast
```

peer
8 2 3
olifant
konijn
lolifant

Uitvoer bij de voorbeeldinvoer

appelr
kleden
komeXX
popenX
kwastX
KN

Voorronde Programmeren 1994

Probleem E - News

NetNews is niet meer weg te denken. Dagelijks worden via een gigantisch computernetwerk berichten verzameld en naar alle andere computers gestuurd. Heb je je ooit afgevraagd via welk algoritme de berichten naar iedere computer worden verstuurd en hoe het netwerk wordt gebruikt (m.a.w. over welke draadjes de berichten worden verstuurd)?

Nee? Wel, zo gebeurt het niet:

Een computernetwerk valt te representeren als een ongerichte graaf (we gaan even van full-duplex verbindingen uit). Elke knoop in de graaf is een computer en elke kant in de graaf is een draad waarover de berichten verstuurd worden. Deze "draden" worden vaak beheerd door bedrijven (zoals de PTT) en deze bedrijven brengen voor het gebruik van de draad bepaalde kosten in rekening. Elke draad kan een ander tarief hebben en daarmee is de graaf dus een gewogen graaf geworden.

Opgave

Een specificatie van een computernetwerk bestaat uit een graaf met de volgende informatie: voor iedere verbinding tussen twee computers een kostprijs voor het aanleggen van deze verbinding. De opgave is nu om gegeven de prijzen voor de verschillende links een zo goedkoop mogelijk netwerk aan te leggen waardoor nog net alle computers met elkaar verbonden zijn.

Beschrijving van de invoer

De invoer bestaat allereerst uit een regel met het aantal problemen dat moet worden opgelost. Daarna komt er per probleem: een regel waarop twee getallen staan; het eerste getal is het aantal computers in het netwerk (h), het tweede getal het aantal verbindingen dat tussen de computers kan worden aangelegd (l). Daarna volgen er l regels met de specificatie van de verbindingen, deze bestaat uit telkens drie getallen: het eerste en tweede getal geven de computers aan waartussen een verbinding ligt, het derde getal de kosten van het aanleggen van die verbinding.

De computers zijn opvolgend genummerd vanaf 1, er zijn maximaal 50 computers en de computers vormen 1 netwerk. Tussen twee computers legt de PTT maximaal 1 verbinding aan.

Beschrijving van de uitvoer

De uitvoer die per probleem gegenereerd moet worden is een getal dat aangeeft wat de totale kosten van het netwerk zijn, als voor de goedkoopste mogelijkheid wordt gekozen waarbij nog net alle computers zijn verbonden, dus nog net 1 netwerk vormen.

Voorbeeldinvoer

```
1
5 8
1 2 1
2 3 5
3 4 2
4 1 1
```


4 2 9
1 5 1
2 5 2
3 5 3

Uitvoer bij de voorbeeldinvoer

5

Uitleg: er is in het voorbeeld gekozen voor de verbindingen tussen computers 1-2, 1-4, 3-4 en 1-5. Met deze verbindingen zijn alle computers nog net bereikbaar binnen het netwerk.

Voorronde Programmeren 1994

Probleem F - Vakantiewerk

Guus moet van zijn vader in de vakantie een week werken, zodat hij eens leert zijn handen uit de mouwen te steken. Omdat zijn vader wil dat Guus zoveel mogelijk doet in deze week, heeft hij een lijst laten maken van al het werk dat er in die week is. Guus mag zelf uitkiezen welke baantjes hij doet, maar hij mag geen werk laten glijpen als dat niet hoeft. Hij mag dus wel een baantje laten zitten wanneer hij daardoor een ander baantje kan doen. M.a.w. zolang er in de voorkeurlijst van Guus nog een baantje bij past, dan is de lijst nog niet compleet.

Guus is door en door lui en wil daarom zo min mogelijk werk doen. Omdat zijn vader ook niet de beroerdste is, zal hij een lading energietabletten bestellen zodat Guus ook 's nachts kan werken. Omdat deze tabletten erg duur zijn, is het van belang precies de juiste hoeveelheid te bestellen. Guus kan, wanneer hij precies weet hoelang hij gaat werken, uitrekenen hoeveel tabletten hij nodig heeft. Omdat Guus ze zelf moet betalen (en bovendien er lui is) laat hij jullie een programma schrijven om uit te zoeken wat de minimale tijd is die hij kan werken onder de gegeven voorwaarden.

Als een nieuw baantje begint op hetzelfde tijdstip waarop een vorig baantje eindigt, dan kan Guus ze nog net allebei doen.

Beschrijving van de invoer

Het aantal instanties van dit probleem staat op de eerste regel. Per instantie eerst een regel met het aantal baantjes in die week. Dan per baantje een regel:

CC DD:DD, D:DD

waarbij door CC de dag van de week wordt aangegeven: Zo, Ma, Di, Wo, Do, Vr en Za. D is een cijfer, waarmee voor de komma het tijdstip waarop het werk begint aangegeven wordt en na de komma de tijd die Guus moet werken voor dit baantje.

De baantjes worden gesorteerd op begintijdstip aangeboden, waarbij de week dus begint op Zo 00:00. Er zijn niet meer dan 50 baantjes in een week.

Beschrijving van de uitvoer

Per instantie een regel met de tijd dat Guus zal moeten werken uitgedrukt in minuten.

Voorbeeldinvoer

```
1
5
Ma 04:10, 2:00
Ma 05:00, 1:00
Ma 07:30, 3:00
Ma 21:00, 2:07
Ma 21:05, 0:15
```

Uitvoer bij de voorbeeldinvoer

255

Voorronde Programmeren 1994

Probleem G - De Ski r

Een ski r zoekt in een skigebied naar de langst mogelijke afdaling. Daartoe raadpleegt hij de hoogtekaart. Hij zoekt op de hoogtekaart naar een route die nergens stijgt of vlak loopt, maar overal daalt, en bovendien zo lang mogelijk is.

Om hem te helpen schrijven we een programma dat, gegeven een hoogtekaart, de lengte van de langst mogelijke dalende route op die kaart geeft. Buren van een punt zijn de punten die er direct naast, boven of onder liggen. Ieder punt heeft dus vier burenen.

Beschrijving van de invoer

Op de eerste regel staat het aantal hoogtekaarten, daarna volgt voor iedere hoogtekaart de grootte van de kaart $x*y$, met daarna de representatie van de hoogtekaart (y regels met x getallen). De hoogtekaart wordt gerepresenteerd door in een rechthoekig coördinatenstelsel steeds de hoogte te geven van de punten met geheeltallige coördinaten.

Grenzen: een hoogte is een getal groter dan 0 en kleiner dan 1000. De afmetingen van de hoogtekaarten zijn maximaal $20*20$.

Voorbeeldinvoer

```
1
4 3
123 127 167 234
 87 235 232 345
121 232 323 343
```

Uitvoer bij de voorbeeldinvoer

7

Voorrunde Programmeren 1994

Probleem H - Chetal

Met behulp van het desktop publishing programma Chetal is het mogelijk om lijnstukken te tekenen in figuren die in je artikel, verslag of boek wilt opnemen. Hiervoor is een speciale instructie beschikbaar, waaraan je als argumenten een beginpunt van een lijnstuk, zijn richtingsvector en zijn lengte in een bepaalde richting kunt meegeven.

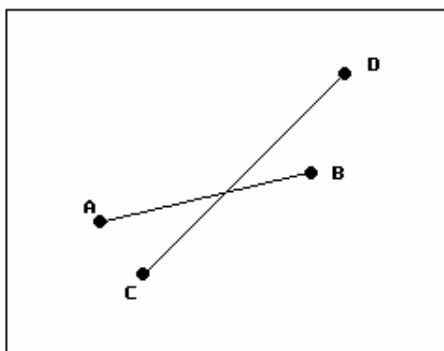
De precieze syntax van deze instructie is

```
\put(x,y){\line(dx,dy){len}}
```

Hierin zijn x en y respectievelijk de x - en y -coördinaat van het gebruikte beginpunt. De richting van het lijnstuk vanaf dit beginpunt wordt aangegeven met de vector (dx,dy) , waarbij dy de verschuiving in de y -richting is die met een verschuiving dx in de x -richting correspondeert. In principe is len de lengte, de absolute waarde dus, van het lijnstuk in de x -richting. Als het beginpunt van het lijnstuk x -coördinaat 5 heeft en de x -coördinaat van het eindpunt is 3, dan is len dus gelijk aan $|3 - 5| = 2$. Alleen als deze lengte 0 zou zijn, dus als het lijnstuk verticaal loopt en $dx=0$, dan wordt voor len de lengte van het lijnstuk in de y -richting genomen. Die is dan tevens gelijk aan de normale euclidische lengte van het lijnstuk. In dit geval mag dy niet ook gelijk zijn aan 0.

Chetal kent, wat deze lijnstukken betreft, echter 1 beperking. De coördinaten van de richtingsvector moeten gehele getallen zijn, waarvan de absolute waarden een grootste gemene deler (ggd) 1 hebben. (De ggd van 2 positieve gehele getallen a en b is gedefinieerd als het grootste gehele getal c zodat c zowel een deler van a als van b is. De ggd van bijvoorbeeld 6 en 15 is 3.) De ggd van 0 en een willekeurig positief geheel getal a is per definitie gelijk aan a . Niet alle lijnstukken die je zou willen hebben, kun je dus met de genoemde instructie tekenen. Je kunt echter wel elk lijnstuk willekeurig dicht benaderen door de juiste argumenten dx en dy aan de instructie mee te geven, uiteraard in combinatie met de juiste waarden voor x , y en len . Deze laatste drie variabelen mogen elke reële waarde aannemen, waarbij len natuurlijk wel positief moet zijn.

Laat de afwijking tussen twee lijnstukken gedefinieerd zijn als de maximale euclidische afstand tussen corresponderende begin- en eindpunten van de verschillende lijnstukken. In de figuur hieronder is de afwijking tussen de twee lijnstukken dus het maximum van de afstanden tussen punt A en punt C en tussen punt B en punt D.



Dan is het de bedoeling om bij een gegeven beginpunt (x_1,y_1) en eindpunt (x_2,y_2) van het gewenste lijnstuk, en een gegeven toegestane afwijking $e > 0$, een Chetal instructie te genereren die het gewenste lijnstuk benadert met een afwijking die kleiner

is dan of gelijk aan e . Verder moeten, om het rekenwerk te vereenvoudigen, zo klein mogelijke getallen dx en dy gebruikt worden, dat wil zeggen: de som van hun absolute waarden moet zo klein mogelijk zijn. Indien er onder deze voorwaarden meerdere mogelijkheden voor (dx,dy) zijn, moet die richtingsvector gekozen worden die van de mogelijkheden, de kleinste afwijking oplevert. Als er dan nog meerdere oplossingen mogelijk zijn, dan moet de steilste richtingsvector genomen worden. Wanneer eenmaal de vector (dx,dy) bepaald is, moeten uiteraard de waarden voor x , y en len zo worden gekozen, dat de uiteindelijke afwijking van het getekende lijnstuk ten opzichte van het gewenste lijnstuk minimaal is.

Beschrijving van de invoer

De invoer bestaat uit een aantal regels, met op elke regel 5 decimale getallen: respectievelijk x_1 , y_1 , x_2 , y_2 en e , gescheiden door spaties. De coördinaten x_1 , y_1 , x_2 en y_2 zijn gegeven in ten hoogste 3 decimalen.

Beschrijving van de uitvoer

Voor elke regel van de invoer een Chetal instructie die het gewenste lijnstuk van (x_1,y_1) naar (x_2,y_2) benadert met een afwijking die kleiner dan of gelijk aan e is. Verder moet aan de eisen uit de beschrijving voldaan zijn. Het punt (x,y) in de uitvoer hoeft niet per se met het punt (x_1,y_1) uit de invoer te corresponderen. Het mag ook overeenkomen met het punt (x_2,y_2) uit de invoer. De waarden x , y en len moeten in drie decimalen nauwkeurig gegeven worden. Als het beginpunt en het eindpunt van het gewenste lijnstuk samenvallen, zodat er niet echt van een lijnstuk sprake is, moet er een lege regel in de uitvoer komen.

Voorbeeldinvoer

```
1 1 4.0 2.0 0.6
2 3 1.5 3.5 0.3
0.356 4.372 5.21 -8.732 0.01
```

Uitvoer bij de voorbeeldinvoer

```
\put (1.000,1.500) {\line (1,0) {3.000}}
\put (1.750,3.000) {\line (0,1) {0.500}}
\put (5.210,-8.732) {\line (-10,27) {4.853}}
```

De eerste regel uit het voorbeeld komt overeen met de onderstaande figuur:

