A Medaillespiegel

Probleem

Op de Olympische Spelen in Athene heeft Nederland een goede prestatie neergezet. Er zijn 22 medailles gehaald (4 goud, 9 zilver en 9 brons) en daarmee staat Nederland op de 14^e plek in de "medaillespiegel".

Deze wordt gemaakt door alle landen eerst te sorteren op het aantal gouden medailles, dan bij gelijk aantal goud op het aantal zilveren medailles en tenslotte op het aantal bronzen medailles. Zo stond Nederland bijvoorbeeld onder Griekenland, dat 5 goud, 2 zilver en 3 brons had en dus in totaal slechts 10 medailles.

Iedereen ziet natuurlijk dat dit niet de goede manier is om de ranglijst op te stellen. Daarom stellen we voor om een ranglijst te maken, door voor ieder land een medaillescore uit te rekenen: vermenigvuldig het aantal goud met een factor A, zilver met een factor B en brons met een factor C en tel dat bij elkaar op. Gegeven de gehaalde medailles door alle landen, hoe moeten die factoren gekozen worden, zodat Nederland zo hoog mogelijk eindigt in het medailleklassement? Wanneer landen op een gelijke score uitkomen, delen ze de hoogste plaats die bij die score hoort. (Dus als land X 10 punten scoort en landen Y en Z 5 punten, dan delen Y en Z de tweede plaats.)

De factoren A, B en C moeten geheeltallig zijn en $100 \geq A > B > C \geq 1$. Verder zal in de lijst met landen een land nooit dubbel voorkomen en "Nederland" zal precies één keer voorkomen.

Invoer

Op de eerste regel één positief getal: het aantal testgevallen.

Daarna per testgeval:

- Een regel met een getal n met $1 \leq n \leq 100$: het aantal landen.
- n regels met daarop een landnaam, bestaande uit minimaal 1 en maximaal 20 hoofd- en kleine letters, gevolgd door drie getallen $0 \leq g, z, b \leq 100$: het aantal gouden, zilveren respectievelijk bronzen medailles gehaald door dat land.

Uitvoer

Per testgeval:

- Een regel met één getal: de best mogelijk haalbare positie van Nederland in het medailleklassement.

Voorbeeld in- en uitvoer

Invoer	Uitvoer
2 3 Nederland 2 4 4 Belgie 4 2 1 Duitsland 3 4 3 20 VerenigdeStaten 25 22 19 China 24 12 10 Australie 14 9 14 Japan 13 6 6 Duitsland 11 10 15 Rusland 9 15 18 Frankrijk 8 10 6 Italie 7 7 8 Roemenie 7 5 2 GrootBrittannie 6 6 9 Oekraine 6 3 7 ZuidKorea 5 10 5 Griekenland 5 2 3 Nederland 4 9 9 Hongarije 3 5 1 Zweden 3 0 1 Turkije 3 0 1 Canada 2 4 1 WitRusland 2 3 6 Polen 2 2 3	2 9

B Spiraal

Probleem

We kijken naar de volgende spiraal van getallen

			−13	−12	−11	
	7	6	5	4	−10	
	8	−2	−1	3	−9	
	9	−3	1	2	−8	18
	10	−4	−5	−6	−7	17
	11	12	13	14	15	16

In de spiraal heeft elk getal 4 buren. Bijvoorbeeld het getal 3 heeft als buren de getallen 4, -9, 2 en -1. De buren zijn dus de getallen die direct grenzen (maar niet diagonaal) aan het gegeven getal. We vragen ons af wat de buren van een bepaald getal zijn.

Invoer

Op de eerste regel één positief getal: het aantal testgevallen.

Daarna per testgeval:

- Een regel met een positief getal $1 \leq n \leq 10^9$.

Uitvoer

Per testgeval:

- Een regel met daarop de buren van het getal n, gescheiden door een spatie en gesorteerd van klein naar groot.

Voorbeeld in- en uitvoer

Invoer	**Uitvoer**

`3`
`1`
`3`
`5`
	`-5 -3 -1 2`
	`-9 -1 2 4`
	`-13 -1 4 6`

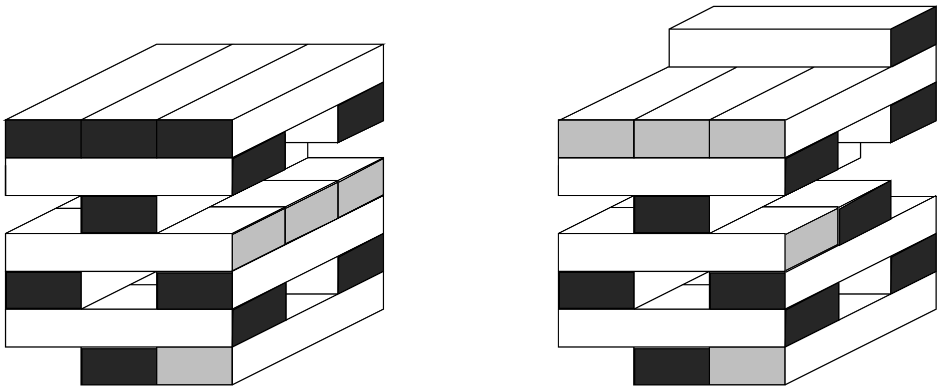
C Jenga

Probleem

"Jenga" is een twee-spelerspel, waarbij je om de beurt een blokje uit een toren moet halen zonder die in te laten storten. Wie de toren in laat storten heeft verloren.

In het begin van het spel staat er een toren, die opgebouwd is uit een aantal lagen, waarbij iedere laag bestaat uit drie langwerpige, balkvormige blokjes. Deze lagen liggen steeds kruislings over elkaar. Het spel wordt gespeeld met twee spelers en de bedoeling is nu dat iedere speler om de beurt een blokje uit een willekeurige (maar niet de bovenste!) laag pakt en daarmee de bovenste laag afbouwt, of als die af is, een nieuwe laag begint. De speler die als laatste succesvol een blokje bovenop legt voordat de toren instort, heeft gewonnen.

Hieronder zie je twee torens zoals die tijdens een spelletje Jenga voor zouden kunnen komen. De blokken die donker zijn gekleurd mogen niet gepakt worden, omdat dan de toren in zou storten of omdat ze op de bovenste laag liggen. Merk op dat je ook niet het laatste blokje uit een laag mag pakken, zodat de rest erboven één laag naar beneden valt. De blokken die licht zijn gekleurd mag je wel pakken. De toren rechts is ontstaan door van de linker toren uit de 4^e laag van onderen het achterste blokje te pakken en bovenop te leggen.



We gaan er in deze opgave vanuit dat de spelers zo behendig zijn, dat ze ieder blokje weg kunnen halen, dat de toren niet laat instorten. Ook gaan we er vanuit, dat een toren alleen instort als er minimaal één laag (en niet de bovenste) is, waarin zich maar één blokje bevindt en dat blokje niet in het midden ligt. Je hoeft dus geen rekening te houden met het feit dat een toren door een ongebalanceerde massaverdeling misschien in zou storten.

Op een dag zit je tegen iemand een potje Jenga te spelen en halverwege het spel vraag je je af of je het spel nog kan winnen. Je weet dat je tegenstander erg slim is en volgens de beste tactiek zal spelen. Je krijgt als invoer een Jenga-toren, zoals die er tijdens het spel bij staat en op dat moment ben jij aan zet. Kun je nog winnen?

Invoer

Op de eerste regel één positief getal: het aantal testgevallen.

Daarna per testgeval:

- Een regel met daarop één geheel getal n met $(2 \leq n \leq 250)$: het aantal lagen in de toren.
- n regels die de toren van bovenste naar onderste laag beschrijven. Op iedere regel staat steeds een string van 3 karakters, die ieder een 'X' (hoofdletter) of een '.' zijn. Dit geeft aan of op de drie posities in de laag van rechts of voor bekeken zich een blokje ('X') bevindt of niet ('.'). Op iedere regel bevindt zich in ieder geval één 'X' en de toren zal niet meteen instorten.

Uitvoer

Per testgeval:

- Eén regel met daarop het woord “winst” als we kunnen winnen en “verlies” als we gaan verliezen.

Voorbeeld in- en uitvoer

Invoer	**Uitvoer**

`2`
`4`
`.X.`
`X.X`
`X.X`
`.X.`
`4`
`.XX`
`X.X`
`XXX`
`XX.`
	`verlies`
	`winst`

D Bierprijzen

Probleem

Het Europees Kampioenschap Programmeren wordt dit jaar weer in Lund gehouden. De trip naar Zweden is erg prijzig en de bierprijzen in Scandinavië zijn hoog. Daarom besluit een aantal deelnemende studenten om hun eigen bier mee te nemen voor onderweg en ook nog wat extra om onderweg of op de plaats van bestemming te verkopen.

Per gereden kilometer nuttigen de studenten tesamen één biertje en dit is van levensbelang! In de kofferbak passen slechts k biertjes, dus het kan zijn dat de bestemming onhaalbaar is, omdat de steden op de route te ver van elkaar verwijderd liggen. Als dit zo is, willen de studenten dat graag van tevoren weten, zodat ze een andere route kunnen plannen. Als ze Lund wel kunnen halen, willen ze graag weten hoeveel winst of verlies ze gaan maken bij optimale in- en verkoop.

Om dit allemaal te bepalen hebben ze uitgezocht hoever de n steden op de route van elkaar verwijderd liggen en wat de lokale bierprijzen zijn (dit is zowel de inkoop- als de verkoopprijs). Help de studenten met hun vragen!

Invoer

Op de eerste regel één positief getal: het aantal testgevallen.

Daarna per testgeval:

- Een regel met twee positieve getallen n en k met $2 \leq n \leq 100$ en $1 \leq k \leq 100$: het aantal steden op de route en de maximale inhoud van de kofferbak.
- Een regel met $n-1$ positieve getallen $1 \leq d_i \leq 100$: de afstanden tussen de steden. d_i is de afstand tussen de i-de en $i+1$-de stad.
- Een regel met daarop n positieve getallen $1 \leq p_i \leq 100$: de bierprijzen in de steden. p_i is de prijs in de i-de stad.

Uitvoer

Per testgeval:

- Een regel met daarop "onmogelijk" als Lund onhaalbaar is of één getal: de beste mogelijke opbrenst (positief bij winst, negatief bij verlies).

Voorbeeld in- en uitvoer

Invoer	Uitvoer
3 2 10 5 10 100 2 10 9 10 10 3 5 4 30 1 2 3	400 -90 onmogelijk

E Lund

Probleem

Naast dat het bier erg duur is in Lund, is er nog een groot probleem: het is een doolhof met veel te veel éénrichtingsverkeerstraten. Het is soms zelfs onmogelijk om zonder verkeersovertredingen op de gewilde bestemming aan te komen. Omdat iedere overtreding een kans op een boete oplevert, willen we natuurlijk zo min mogelijk overtredingen maken.

Gelukkig hebben we een kaart van Lund om ons hierbij te helpen: hierop is te zien dat het een rechthoekige stad is van l kruispunten lang en b kruispunten breed. Op een gedeelte van de kruispunten geldt een verplichte rijrichting, namelijk Noord, Zuid, Oost of West. Dit wordt op de kaart aangegevenmet respectievelijk 'N', 'Z', 'O' of 'W'. Er zijn ook kruispunten met geen verplichte rijrichting en die zijn aangegeven met '.'. We beginnen linksboven op de kaart en onze bestemming is rechtsonder. Het is niet toegestaan Lund te verlaten door van de kaart af te lopen en verder verplicht geen enkel kruispunt je dat te doen.

Invoer

Op de eerste regel één positief getal: het aantal testgevallen.

Daarna per testgeval:

- Een regel met twee positieve getallen b en l met $1 \leq b, l \leq 100$: de breedte en lengte van Lund.
- b regels met daarop l karakters, 'N', 'Z, 'O', 'W' of '.': de al dan niet verplichte richting op het kruispunt.

Uitvoer

Per testgeval:

- Een regel met daarop het minimale aantal te maken verkeersovertredingen.

Voorbeeld in- en uitvoer

Invoer	Uitvoer
1 4 5 ZOZ.W N.OW. OWOZZ ..NOW	2

F Blaadjes op de rails

Probleem

De herfst is weer begonnen en de directie van de Nederlandse Spoorwegen voorziet problemen: weldra zullen er weer bladeren van de bomen vallen, die op de rails terechtkomen. Dit zal weer tot grote vertragingen kunnen leiden en er zelfs voor zorgen dat bepaalde stukken spoor helemaal onbruikbaar worden.

Dit laatste zorgt er vaak voor dat reizigers een omweg moeten maken om op hun plaats van bestemming aan te komen, maar soms is het zelfs zo erg dat reizigers niet eens meer op hun bestemming aan kunnen komen! Dit is natuurlijk onacceptabel en daarom heeft de directie besloten om nieuwe stukken spoor tussen bepaalde stations te bouwen, op zo'n manier dat als er één stuk spoor uitvalt, je nog altijd tussen ieder paar stations kunt reizen.

Om de kosten te drukken, willen ze echter zo weinig mogelijk extra stukken spoor bouwen. Wat is, gegeven het bestaande spoornetwerk, het minimale aantal extra benodigde stukken spoor?

Invoer

Op de eerste regel één positief getal: het aantal testgevallen.

Daarna per testgeval:

- Een regel met twee positieve getallen n en m, met $1 \leq n \leq 100$ en $1 \leq m \leq 10^4$: het aantal stations en het aantal stukken spoor. Met het bestaande spoornetwerk is het mogelijk tussen ieder paar stations te reizen. (Dit kan direct of indirect zijn.) Er zijn geen stukken spoor van een station naar zichzelf en ook zijn er tussen een tweetal stations niet meerdere stukken spoor.
- m regels met daarop twee getallen tussen 1 en n: de nummers van de stations die door een spoor verbonden worden.

Uitvoer

Per testgeval:

- Een regel met daarop het minimaal aantal extra te bouwen stukken spoor.

Voorbeeld in- en uitvoer

Invoer	Uitvoer
3 4 3 1 2 2 3 3 4 3 3 1 2 1 3 3 2 4 3 1 2 1 3 1 4	1 0 2

G Simulatie van een jurysysteem

Probleem

Op ieder programmeerkampioenschap worden problemen ingestuurd door de deelnemers, die dan door de jury getest worden. De jury heeft hiervoor niet één, maar meerdere computers waarop de oplossingen getest worden.

Het systeem werkt als volgt: alle inzendingen die binnenkomen, worden achteraan een wachtrij geplaatst. Wanneer een computer niets te jureren heeft, wordt de eerste inzending uit de wachtrij genomen en aan die computer toebedeeld.

Iedere inzending heeft een bepaalde tijd die het duurt, voordat de oplossing berekend is, maar zoals bekend heeft de jury ook een tijdslimiet ingesteld: zodra die bereikt wordt, wordt het testen van de inzending meteen beëindigd.

Op tijdstip 0 begint de wedstrijd en is de wachtrij leeg. Gevraagd is het tijdstip wanneer alle inzendingen getest zijn. Verder is gegarandeerd, dat er geen twee oplossingen op exact hezelfde tijdstip binnenkomen.

Invoer

Op de eerste regel één positief getal: het aantal testgevallen.

Daarna per testgeval:

- Een regel met één positief getal m met $1 \leq m \leq 100$: het aantal jurycomputers.
- Een regel met één positief getal t met $1 \leq t \leq 100$: de tijdslimiet van de jury.
- Een regel met één niet-negatief getal n met $0 \leq n \leq 1000$: het aantal inzendingen.
- n regels met per regel twee niet-negatieve getallen s, d met $0 \leq s \leq 1000$ en $1 \leq d \leq 10^6$: het tijdstip waarop de inzending binnenkomt, respectievelijk hoelang deze inzending nodig heeft om de oplossing te berekenen.

Uitvoer

Per testgeval:

- Een regel met daarop een integer: het tijdstip waarop alle inzendingen getest zijn.

Voorbeeld in- en uitvoer

Invoer	Uitvoer
1 2 10 3 2 20 3 5 7 8	16

H Voetbal

Probleem

Na de blamage van afgelopen zomer heeft de kersverse bondscoach Marco van Basten besloten om het radicaal anders aan te pakken. Zijn nieuwe methode luidt als volgt:

Het speelveld is opgedeeld in n verschillende posities en op ieder van die posities moet hij een speler opstellen. Hij heeft voor al zijn m mogelijke spelers bepaald wat zijn kwaliteit is op iedere veldpositie en hij wil precies die n spelers op de n veldposities zetten, zodat de totale kwaliteit (dat is de som van de individuele kwaliteiten) maximaal is.

Wat is de maximaal mogelijke kwaliteit van het team?

Invoer

Op de eerste regel één positief getal: het aantal testgevallen.

Daarna per testgeval:

- Een regel met twee positieve getallen n en m met $1 \leq n \leq 15$ en $n \leq m \leq 50$: het aantal veldposities en het aantal mogelijke spelers.
- m regels met ieder n niet-negatieve getallen k_{ij} met $0 \leq k_{ij} \leq 1000$: de kwaliteit van de i-de speler op de j-de veldpositie.

Uitvoer

Per testgeval:

- Een regel met daarop de maximale mogelijke kwaliteit van het team.

Voorbeeld in- en uitvoer

<table>
<tr><th>Invoer</th><th>Uitvoer</th></tr>
<tr><td><pre>
2
2 3
2 0
1 1
0 2
3 3
20 10 0
20 0 10
0 20 10
</pre></td><td><pre>
4
50
</pre></td></tr>
</table>