

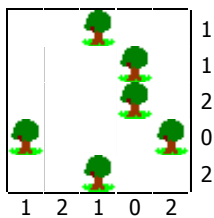
Opdracht 0. Tentje-boompje puzzels. Deel II.

Deze beschrijving is afkomstig van www.puzzelhulp.nl

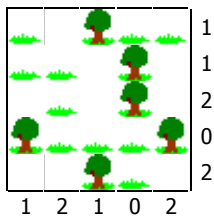
Een wat zeldzamer puzzel heet Tentje-Boompje. Het idee is dat het puzzelveld een camping voorstelt. Op de camping staan een flink aantal bomen. Er wordt verteld dat er een aantal kampeerders hun tent wil opslaan. De puzzelaar speelt de campingbeheerder die iedereen een plekje wil geven.

Zo'n plek moet aan twee eisen voldoen. Er moet een boom direct naast de tent staan (schuin telt niet) en er mag geen andere tent naast staan, ook niet schuin. Ten slotte moeten in elke rij en elke kolom precies zoveel staan als er achter staat.

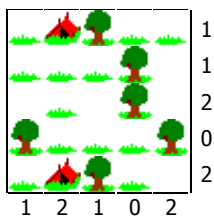
Stel dat een kleine camping er als volgt uit ziet:



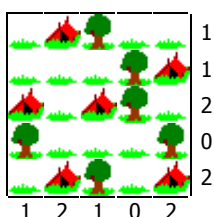
De eerste stap is makkelijk. De vierde kolom en de vierde rij bevatten geen tentjes, dus die kunnen we afstrepen. Ook de vakjes die te ver van een boom staat, of er diagonaal naast staan, strepen we af. Uiteraard kunt u op papier gewoon een stip in zo'n vakje zetten.



De eerste rij en de tweede kolom kunnen nu worden afgemaakt. Er zijn immers evenveel open plaatsen als dat er tentjes moeten staan. Let erop dat bij elk tentje in de acht vakjes er omheen geen andere tentjes kunnen staan.



De rest kan nu eenvoudig ingevuld worden.



Voor deze opgave ga je een aantal aspecten van zo'n tentje-boompje onderzoeken. De opgave bestaat uit twee delen; deel I ontvang je via de mail op vrijdagmiddag, deel II ontvang je in Leiden of Twente op zaterdagochtend.

Invoer:

Invoer is een bestand **diagram.in**.

Op de eerste regel staat een getal R , met $2 \leq R \leq 12$, dat het aantal regels van het diagram aangeeft. Op de tweede regel staat een getal K , met $2 \leq K \leq 12$, dat het aantal kolommen van het diagram aangeeft.

Vervolgens zijn er R regels die de plaatsing van de bomen beschrijven. Op iedere regel staan K tekens, een punt als er geen boom staat of een hoofdletter B als er wel een boom staat, en nog een cijfer C , met $0 \leq C \leq 6$, dat aangeeft hoeveel tenten er op deze regel van het diagram geplaatst moeten worden. Daarna is er nog een regel met K cijfers C , weer elk met $0 \leq C \leq 6$, waarmee het aantal te plaatsen tenten per kolom wordt aangegeven.

Voorbeeld bestanden:

<pre> diagram0.in (zie ook de afbeelding van bladzijde 1) 5 5 ..B..1 ...B.1 ...B.2 B...B0 ..B..2 12102 </pre>	<pre> diagram1.in 10 10 ..B.B...B.2 B.....2 ...B....B.2 B...B....2 ...B....B2 .B.....B..2 ...B.B...B2 .B.....2B...1 .B.B....B.3 3223112222 </pre>
---	---

Deze bestanden worden bij de voorbeelden bij alle opgaven als invoer gebruikt.

Voorbeeldbestanden en testen:

Er zijn bestanden **diagram0.in**, **diagram1.in** tot en met **diagram5.in** beschikbaar waarmee je je programma kunt uitproberen.

Er is een batchfile **test0.bat**, die je kunt gebruiken op de volgende manier:

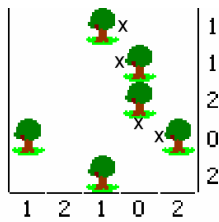
```
test0 nio0a diagram0.in
```

Met deze opdracht test je het programma **nio0a** (of op deze plaats één van je andere programma's), waarbij vooraf eerst de invoer uit **diagram0.in** (of op deze plaats één van de andere bestanden) naar het bestand **diagram.in** wordt gekopieerd. Je zult dan zelf moeten controleren of het programma binnen de tijdlimiet stopt en de goede uitvoerfile maakt.

Opdracht 0 deel II. Taakoverzicht.

Onderdeel	Programma	Uitvoer	Tijdlimiet per test	Aantal testen	Punten per test	Totaal
0D	nio0d	0d.uit	1 sec	5	4	20
0E	nio0e	0e.uit	1 sec	5	6	30

Opdracht 0D: Aan welke kant kunnen de tenten geplaatst?



Wanneer je het diagram van het voorbeeld bekijkt, zul je zien dat op sommige velden geen tenten kunnen worden geplaatst omdat te veel velden er door worden geblokkeerd. Zo kan er rechts van de boom op de bovenste rij geen tent worden geplaatst omdat er dan voor een tent naast de boom op de tweede rij geen plaats meer overblijft (dat heeft dus nog niets te maken met de voorwaarde dat er in de hele kolom geen tent mag worden geplaatst). Aan de kant van de kruisjes mag geen tent worden neergezet.

Schrijf een programma `nio0d` dat een bestand `diagram.in` inleest. Uitvoer is een bestand `0d.uit` dat bestaat uit een regel voor iedere boom uit de invoer. Op die regel staan één tot vier letters, die aangeven aan welke kant van de betreffende boom een tent die bij die boom hoort kan worden geplaatst zonder dat daardoor andere bomen geen tent naast zich kunnen krijgen. De letters zijn een B voor een tent boven de boom, een R voor een tent rechts van de boom, een O voor een tent onder de boom en een L voor een tent links van de boom, in die volgorde. Er zijn geen bomen in de invoer waar niet tenminste één tent naast kan worden geplaatst.

De bomen worden in de uitvoer op dezelfde volgorde afgehandeld als dat ze in de invoer voorkomen (regel voor regel, en per regel van links naar rechts).

Voorbeeld:	OL	ROL
	BR	ROL
	RL	ROL
	BRO	BRO
	BO	BROL
	BRL	ROL
		BRO
		BROL
		BROL
		BOL
		BRL
		BROL
		BROL
		BROL
		BOL
		BRL
		BROL
		BRL
		BRL
		BRL

Opdracht 0E: De oplossing.

Schrijf een programma `nio0e` dat een bestand `diagram.in` inleest. Uitvoer is een bestand `0e.uit` dat bestaat uit R regels van K tekens. Als op een vakje een boom staat, staat er een B aangegeven. Als er op een vakje geen tent kan staan, staat er een G (van gras). Als er wel een tent staat, staat er een T aangegeven.

Voorbeelden:

GTBGG	TGBGBGGTBG
GGGTB	BGTGTGGGGG
TGTBG	TGGBGGGGBT
BGGGB	BGGTGBTGGG
GTBGT	GTGBGGGGTB
	GBGTGGTBGG
	GTGBGBGGTB
	GBGTGTGGGG
	GGGGGGBTGG
	TBTBGGGGBT