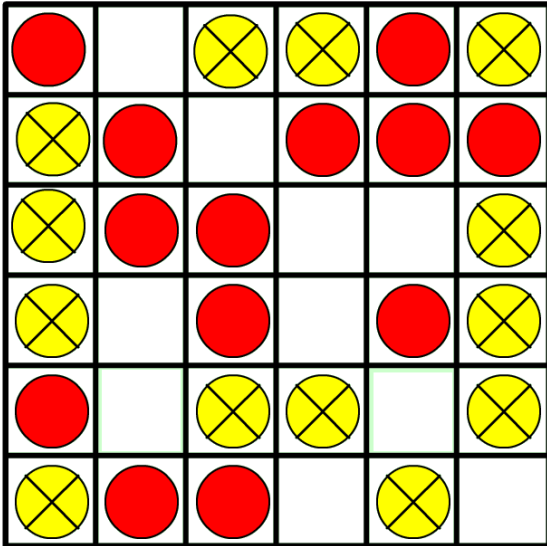


## Opgave 2. Naar orde streven (of juist niet)

Opgave 2 is een vervolg van Opgave 1 waarin de spelregels van het spel Chaos en Orde zijn uitgelegd. In opgave 2 mag je er van uitgaan dat beide spelers samenwerken naar hetzelfde doel. Hun en ook jouw opdracht is om de resterende stenen zo neer te leggen dat ze na afloop van het spel een zo hoog mogelijke score hebben behaald (of bij opgave 2D een zo laag mogelijke).

### Opgave 2a en 2b. Een gedeeltelijk gevuld bord aanvullen



Voor zowel opgave 2a als 2b moet je een programma schrijven dat een gedeeltelijk gevuld spelbord inleest uit een tekstbestand. Dit bestand heet `board.in` en bestaat uit zes regels; op iedere regel staan zes tekens, die de verschillende stenen op het bord weergeven. Voor de stenen worden de hoofdletters `O` en `X` gebruikt. Als er geen steen is geplaatst wordt dat aangegeven met een `.` (een punt).

Voor de invoer geldt altijd dat er van beide soorten stenen evenveel op het bord zijn geplaatst. Dat moet uiteraard ook het geval zijn bij de eindstelling die je programma moet bepalen.

Voorbeeld:

```
O . XXOX
XO . OOO
XOO . .X
X . O .OX
O . XX .X
XOO . X .
```

### Opgave 2a: Hoogste vlakke score

Schrijf een programma `nio2a` dat een bestand `board.in` inleest. Je programma berekent voor het spelbord in de invoer de eindstelling met de hoogst mogelijke **vlakke score**. Uitvoer is een tekstbestand `nio2a.uit` met deze eindstelling, net zo opgebouwd als de invoer voor opgave 1.

Voorbeeld:	Score per rij	Totaalscore
OXXXXX	4	
XOOOOO	16	
XOOOOX	9	
XOOOOX	9	
OXXXXX	16	
XOOXXX	1+4	59

### Opgave 2b: Hoogste totale score

Schrijf een programma `nio2b` dat een bestand `bord.in` inleest. Je programma berekent voor het spelbord in de invoer de eindstelling met de hoogst mogelijke **totale score**. Uitvoer is een tekstbestand `nio2b.uit` met deze eindstelling, net zo opgebouwd als de invoer voor opgave 1.

Voorbeeld:

```
OXXXXX
XOOOOO
XOOOOX
XOOXOX
OXXXXX
XOOXXX
```

De totale score van deze uitvoer is 110 (namelijk horizontaal  $4+16+9+1+1+9+1+4=45$ , verticaal  $4+16+4+1+4+9+1+9=48$  en diagonaal  $9+4+4=17$ ).

Tijdlimiet:

Je programma's voor de opgaven 2a en 2b krijgen maximaal 5 seconden de tijd om een oplossing te produceren. Voor testgevallen waarin minimaal 16 stenen zijn geplaatst verwachten we de optimale (hoogst mogelijke) oplossing. Voor testgevallen met minder dan 16 stenen op het bord kun je ook met een niet helemaal optimale oplossing nog een deel van de punten krijgen.

### Opgave 2c en 2d. Nu met een leeg bord van 10 bij 10

In dit gedeelte van de opgave is het spelbord 10 bij 10 vakjes. Hierbij wordt alleen naar de **totale score** gekeken.

We vragen je om twee eindstellingen te produceren, één met een maximale orde, dus een zo groot mogelijke totale score, de andere met een minimale orde (oftewel een maximale chaos), dus met een zo laag mogelijke totale score.

Je hoeft voor deze opgave dus geen programma's in te leveren, zelfs niet eens te gebruiken. Je moet alleen de gevraagde bestanden maken.

### Opgave 2c: Grootst mogelijke totale score

Maak een bestand `nio2c.uit` dat bestaat uit 10 regels van elk 10 tekens, die de verschillende stenen op het bord aangeven. Voor deze stenen worden de hoofdletters `O` en `X` gebruikt. De totale score voor dit spelbord moet zo groot mogelijk zijn.

### Opgave 2d: Kleinst mogelijke totale score

Maak een bestand `nio2d.uit` dat bestaat uit 10 regels van elk 10 tekens, die de verschillende stenen op het bord aangeven. Voor deze stenen worden de hoofdletters `O` en `X` gebruikt. De totale score voor dit spelbord moet zo klein mogelijk zijn.

## Jurering

Opgave	Aantal testgevallen	Minstens 16 stenen op het bord?	Punten per testgeval	Gedeeltelijke score mogelijk?	Maximum te halen
2a	5	ja	6	nee	30
2b	5	ja	6	nee	30
	3	nee	5	ja	15
2c	1	n.v.t.	10	ja	10
2d	1	n.v.t.	15	ja	15

Zoals je ziet kunt je bij de opgaven 2b, 2c en 2d ook punten krijgen voor oplossingen die in de buurt van de beste oplossing liggen. Bij 2b en 2c krijg je in dat geval al punten als je minstens de helft van de totale score van de beste oplossing hebt gevonden. Bij 2d krijg je al punten als je hoogstens twee maal zo hoog uitkomt als de totale score van de beste oplossing.