

De eerste ronde Nederlandse Informatica Olympiade 2014-2015

De informatica olympiade is een wedstrijd voor leerlingen uit het voortgezet onderwijs in Nederland. Het is een wedstrijd die bestaat uit drie ronden. In de derde ronde wordt bepaald wie Nederland mogen vertegenwoordigen op de Internationale Informatica Olympiade in juli 2015 in Kazachstan.



De eerste ronde

De eerste ronde van de Nederlandse Informatica Olympiade bestaat dit jaar uit 12 opgaven. Deelnemers die tenminste 200 punten halen krijgen een certificaat. Heb je tussen de 200 en 399 punten dan staat op het Certificaat de vermelding Brons. Tussen de 400 en 599 punten de vermelding Zilver en bij meer dan 600 punten de vermelding Goud.

De beste 100 leerlingen worden uitgenodigd voor de tweede ronde, die in maart 2015 wordt gehouden op de Universiteit Twente. Voor deelname aan die tweede ronde moet je wel minstens 200 punten hebben gehaald.

Voor de beste deelnemer van klas 1 en 2, de beste deelnemer van klas 3 en 4 en de beste deelnemer van klas 5 en 6 is een aparte prijs beschikbaar.

Er zijn vier soorten opgaven:

Soort	Omschrijving	Aantal	Punten per opgave	Totaal te halen
A	Inleidende opgaven	5	40	200
B	Theoretische opgaven	4	50	200
C	Programmeeropgaven	2	100	200
D	Een spel programmeren	1	100	100

Om deel te kunnen nemen moet je een account maken op submit.informaticaolympiade.nl

Bij de eerste keer aanmelden moet je enkele gegevens aanleveren, die wij nodig hebben om de olympiade goed te kunnen organiseren. Als je deze gegevens niet wilt of kunt aanleveren, kun je helaas niet deelnemen. Je verklaart in de laatste stap dat je de gegevens naar waarheid hebt ingevuld; daarna staat deelname voor je open. Als je van vorige jaren al een account hebt, zul je de gegevens ook eventueel eerst moeten aanvullen voor je verder kunt werken in het systeem.

Je kunt je uitwerkingen uploaden naar submit.informaticaolympiade.nl wanneer je in het systeem bent ingelogd. In het systeem kun je ook een voorbeeldopgave insturen om uit te proberen hoe het werkt. De opgaven worden meteen geheel of gedeeltelijk nagekeken, voor de rest van de uitslag zul je moeten wachten op het resultaat. Je uitwerkingen voor de opgaven A, B en C moeten uiterlijk 15 januari worden geüpload. Op 17 januari wordt de eerste ronde gejureerd en kort daarna worden de uitslagen gepubliceerd.

Voor de spelopgave, opgave D, moet je je aanmelden op nio3.codecup.nl en kun je via die site ook je programma uploaden. De deelnemende programma's die meewerken met het jurysysteem komen op 17 januari 2015 tegen elkaar uit in een toernooi van te volgen is op nio3.codecup.nl en de winnaar wint de jaarlijkse Windesheim Digitalisprijs van 250 euro.

Voor alle opgaven geldt dat je er van uit mag gaan dat je programma's alleen correcte invoer aangeboden krijgen.

Opgaven A1 tot en met A5

Deze opgaven zijn vooral bedoeld voor leerlingen die beginnen met programmeren. Vanuit de olympiade bieden we lesmateriaal aan om te beginnen met programmeren met Python. Dat is de cursus CS Circles van de Universiteit van Waterloo in Canada: <http://cscircles.cemc.uwaterloo.ca> . Er is een Nederlandse vertaling beschikbaar: <http://cscircles.cemc.uwaterloo.ca/0-nl/> . In de tekst van die Nederlandse vertaling staat vanaf 10 oktober aangegeven wanneer je toe bent aan de volgende opgave van de eerste ronde.

Opgaven B1 tot en met B4

De B-opgaven zijn theoretisch van aard. Er is geen voorkennis voor nodig en je hoeft ook niet te kunnen programmeren

Deze opgave kun je één voor één downloaden uit het inzendsysteem. De opgave wordt speciaal voor jou gemaakt en jij moet het antwoord op de opgave die je vanuit het systeem krijgt inleveren. Het heeft dus geen zin om de antwoorden van iemand anders te gebruiken en die in te zenden.

Als je binnen een week het goede antwoord instuurt krijg je 50 punten per opgave. Voor iedere dag later gaat er één punt van je score af. Inzendingen na 15 januari 2015 zullen niet worden verwerkt.

Als je een verkeerd antwoord hebt gegeven, verlies je 10 punten. Ja mag dan de volgende dag een nieuwe poging wagen.

Het gaat bij al deze opgaven om korte antwoorden, een getal of een korte tekst, die je op de betreffende pagina van het inzendsysteem kunt invoeren. Als je je antwoord hebt bevestigd, krijg je meteen je score te zien.

Je mag allerlei hulpmiddelen gebruiken om de opgave op te lossen. Je zou er bijvoorbeeld een computerprogramma bij kunnen schrijven. Noodzakelijk is dat echter niet. Als voorbereiding op het vervolg van de informatica olympiade is het wel een mooie uitdaging om na te gaan hoe je een programma zou kunnen schrijven dat dit probleem, of problemen die er op lijken, kunt oplossen.

Opgaven C1 en C2

Dit zijn wat complexere opgaven waarmee je een probleem moet oplossen door het schrijven van een computerprogramma. Die programma's lezen invoer van standard input (het toetsenbord) en schrijven uitvoer naar standard output (het beeldscherm). Je programma moet zich daarbij precies houden aan de beschrijvingen van de opdracht. Je programma krijgt een aantal testgevallen voorgeschiedeld en voor ieder testgeval kun je punten krijgen.

Opgave D

Bij deze opgave moet je een programma schrijven dat het spel Capture-Go kan spelen. De programma's spelen op 17 januari een toernooi tegen elkaar. Om deel te kunnen nemen moet je programma kunnen samenwerken met onze jurysoftware; voor details verwijzen we naar nio3.codecup.nl

De CodeCup

De Stichting Nederlandse Informatica Olympiade organiseert nog een wedstrijd waarbij een spel moet worden geprogrammeerd. Daaraan mag iedereen deelnemen; de afgelopen jaren hebben we deelnemers gehad uit meer dan twintig verschillende landen. Zie voor deze wedstrijd

www.codecup.nl

Opgave A1. Kerstboom

Schrijf een programma dat een getal N inleest van standard input. Het programma schrijft naar standard output een “kerstboom” van $N+1$ regels van elk $2N-1$ tekens, in een patroon zoals te zien is in het voorbeeld hieronder. De eerste N regels vormen een driehoekig patroon. Op de laatste regel staat één sterretje in het midden van de regel.

Voorbeeld

Invoer: 7

Uitvoer:

```
-----*-----
-----**-----
-----***-----
-----****-----
-----*****-----
-----*****-----
-----*****-----
*****
-----*-----
```

Randvoorwaarde: N is een positief geheel getal, met $1 < N < 41$

Opgave A2. Oneven kwadraten

Schrijf een programma dat van standard input gehele niet-negatieve getallen inleest, tot er een 0 wordt ingelezen. Het programma schrijft één regel uitvoer naar standard output; daarop staat de som van de kwadraten van de oneven getallen in de invoer.

Voorbeeld

Invoer: 12
 9
 6
 3
 0

Uitvoer: 90

Randvoorwaarde: De invoer bestaat uit maximaal 100 getallen. Elk getal in de invoer is niet groter dan 1000.

Opgave A3. Je naam

Schrijf een programma dat een naam inleest van standard input.

Het programma schrijft vier regels uitvoer naar standard output.

- Op de eerste regel staat uit hoeveel tekens de invoer bestaat.
- Op de tweede regel staat hoeveel hoofdletters in de invoer voorkomen.
- Op de derde regel staat hoeveel verschillende tekens in de invoer voorkomen.
- Op de vierde regel staat de naam, maar dan achterstevoren weergegeven.

Voorbeeld:

Invoer: Anne van der Boom

Uitvoer: 17
 2
 11
 mooB red nav ennA

Randvoorwaarde: In de invoer staan maximaal 60 tekens.

Opgave A4. Speldiagram

Schrijf een programma dat vier regels van vier tekens inleest van standard input. Iedere regel bestaat uitsluitend uit de tekens 0 en 1. In de hele invoer komt precies 8 keer een 0 voor, en 8 keer een 1.

Je programma telt horizontaal en verticaal hoeveel aaneengesloten rijtjes er zijn van 2, 3 of 4 dezelfde tekens achter elkaar. Je programma schrijft drie regels naar standard output.

Op de eerste regel staan de aantallen rijtjes in de invoer van twee aaneengesloten nullen en enen, gescheiden door een spatie.

Op de tweede regel staat de aantallen rijtjes in de invoer van drie aaneengesloten nullen en enen, gescheiden door een spatie.

Op de derde regel staan de aantallen rijtjes in de invoer van vier aaneengesloten nullen en enen, gescheiden door een spatie.

Voorbeeld

Invoer: 0111
 0101
 1001
 0010

Uitvoer: 5 1
 0 2
 0 0

Opgave A5. Verkiezing

Bij een vergadering moet er een nieuwe voorzitter worden gekozen. Er zijn verschillende kandidaten. Alle deelnemers aan de vergadering mogen op één van de kandidaten stemmen. Als een kandidaat meer dan de helft van de stemmen heeft gekregen is hij verkozen. Anders valt de kandidaat met de minste stemmen af en wordt er opnieuw gestemd; als twee of meer kandidaten met hetzelfde aantal stemmen eindigen valt degene met het hoogste volgnummer af. Deze procedure wordt herhaald tot er een voorzitter is gekozen.

Alle deelnemers aan de vergadering hebben hun voorkeurslijst al gemaakt. Ze stemmen op hun favoriete kandidaat, als die nog in de strijd is. Anders kiezen de eerst beschikbare uit hun voorkeurslijst.

Schrijf een programma dat een aantal regels inleest van standard input.

- Op de eerste regel staat het aantal K aan kandidaten voor voorzitter.
- Op de tweede regel staat het aantal D aan deelnemers aan de vergadering.
- Op de volgende D regels staan de voorkeurslijsten van de verschillende deelnemers, in de vorm van de getallen 1 tot en met N in de volgorde van hun voorkeur. De getallen zijn gescheiden door een spatie.

Je programma schrijft één regel uitvoer naar standard output, met daarop het nummer van de kandidaat die wordt verkozen tot voorzitter.

Voorbeeld

```
Invoer:  5
         10
         1 2 3 4 5
         2 4 3 5 1
         1 3 5 2 4
         4 2 5 3 1
         1 4 2 5 3
         2 3 1 5 4
         1 2 3 5 4
         2 4 1 3 5
         1 5 4 3 2
         5 4 3 2 1
```

```
Uitvoer:  1
```

Toelichting bij het voorbeeld:

- In de eerste ronde valt kandidaat 3 af. Er verandert niet aan de stemming.
- In de tweede ronde valt kandidaat 5 af. De stem van deelnemer 10 gaat nu naar 4.
- In de derde ronde valt kandidaat 4 af. De stemmen voor haar gaan nu naar kandidaat 2.
- In de vierde ronde krijgen 1 en 2 even veel stemmen. Dus valt kandidaat 2 af en wint 1.

Randvoorwaarden: Er geldt $2 < K < 20$ en $2 < D < 1000$.

Opgave B1 tot en met B4: Download deze van submit.informaticaolympiade.nl

Toelichting:

Opgave B1 t/m B4 zijn theoretische opgaven. Er is geen voorkennis voor nodig en je hoeft ook niet te kunnen programmeren.

Ze doen denken aan opgaven uit de Beverwedstrijd.

Opgave C1. Waslijnen

Je hebt een aantal waslijnen tot je beschikking en een aantal gewassen kledingstukken. Je wilt die kledingstukken zo ophangen dat je zo weinig mogelijk waslijnen nodig hebt.

Schrijf een programma dat invoer leest van standard input.

- Op de eerste regel staat een getal dat de breedte B van elk van je waslijnen aangeeft.
- Op de tweede regel staat het aantal kledingstukken N dat je wilt ophangen.
- Vervolgens komen er N regels, elk met de breedte van een kledingstuk.

Je programma schrijft als uitvoer het aantal waslijnen dat minimaal nodig is om deze kledingstukken op te hangen, zodat kledingstukken niet over elkaar hangen en wel de volle breedte krijgen die ze nodig hebben.

Voorbeeld

```
Invoer:      200
              10
              100
              70
              35
              140
              45
              77
              121
              17
              91
              83
```

```
Uitvoer:      4
```

Randvoorwaarden: Er zijn twee soorten testgevallen.

Voor de eerste groep geldt $5 < B < 1000$ en $3 < N < 20$.

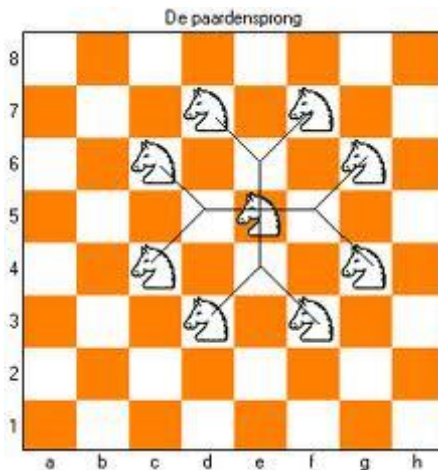
Voor de tweede groep geldt $5 < B < 65$ en $20 < N < 40$.

Geen enkel kledingstuk is breder dan de waslijn.

Er geldt voor deze opgave een tijdlimiet van 2 seconden.

Opgave C2. Springen op een schaakbord

Op een schaakbord (8 bij 8 vakjes) is het mogelijk met een paard (dat beweegt in één beurt één horizontaal en twee verticaal of net andersom, zie plaatje) in 64 stappen het hele bord rond te gaan. Zie de afbeelding hieronder.



Voorbeelden van een paardensprong vanaf veld e5.

1	4	53	13	55	6	43	20	8
32	17	2	5	38	19	56	7	7
3	64	15	54	31	42	21	44	6
16	51	28	39	34	37	8	57	5
63	14	35	32	41	30	45	22	4
30	27	40	29	36	33	58	9	3
18	62	25	48	11	60	23	46	2
26	49	12	61	24	47	10	59	1
a	b	c	d	e	f	g	h	

Voorbeeld van een rondgang van een paard. Het begint op veld a8 en kan in totaal alle 64 vakjes langs. Van het 64^e vakje kun je zelfs weer naar a8 springen.

Als het schaakbord andere afmetingen zou hebben, is het niet per se mogelijk om het hele bord rond te springen. Op een bord van 7 bij 9 kun je wel alle vakjes bereiken, maar niet terug naar het beginvak. En op een bord van 3 bij 3 kun je het middenvakje nooit bereiken.

Als het paard anders zou bewegen is het ook niet altijd mogelijk om het hele bord rond te springen. Met een paard dat één horizontaal en drie verticaal (of andersom) zou springen, zou je niet alle vakjes kunnen bereiken, waar je ook zou beginnen.

Bij deze opgave moet je een programma schrijven dat het “paard” links boven op het bord laat beginnen en dan zo veel mogelijk zetten laat doen, zonder ooit weer op hetzelfde vakje uit te komen. De vakjes worden aangeduid zoals op het schaakbord, dus de kolommen heten a en hoger (maximaal tot en met j) en de rijen heten 1 en hoger (maximaal tot en met 9).

Schrijf een programma dat twee regels invoer leest van standard input.

- Op de eerste regel staan de afmetingen van het bord, twee getallen, gescheiden door een spatie.
- Op de tweede regel staan de stappen in beide richtingen van het paard, twee getallen S met $0 \leq S \leq 5$, gescheiden door een spatie.

Je programma voert een aantal regels uit naar standard output. Op het eerste vakje staat het aantal vakjes N dat je op je mogelijke rondgang zou kunnen bereiken. Daarna volgen N regels, elk met de naam van een vakje op het bord. Deze vakjes geven de tocht van het paard over het bord aan. Het eerste vakje is altijd het vakje links boven op het bord.

Voorbeeld

Invoer: 10 2
 1 1

Uitvoer: 10
 a2
 b1
 c2
 d1
 e2
 f1
 g2
 h1
 i2
 j1



Bij veel invoer zijn er verschillende oplossingen mogelijk. Je programma moet één oplossing geven.

Voor deze opgave geldt een tijdlimiet van 2 seconden.

In de helft van de testgevallen heeft het bord een oppervlakte van niet meer dan 20 vakjes.

Voor elk testgeval geldt dat je $\frac{\text{<jouw_antwoord>}}{\text{<beste_antwoord>}} \cdot 10$ punten krijgt, als je een geldige zettenreeks hebt uitgevoerd; er zijn in totaal 10 testcases.