**Werkblad Sorteren van kaarten deel I**

**Doel**: een set kaarten sorteren

**Wat heb je nodig**: een set van 9 kaarten van dezelfde kleur, bijv harten 2 t/m 10.

Je werkt in teams van 3 leerlingen. Verdeel de rollen:

* Een leerling is de **vergelijker**: hij/zij krijgt steeds twee speelkaarten te zien en wijst aan welke lager is.
* Een leerling is de **uitvoerder**: hij/zij voert het programma uit. Deze leerling gaat de kaarten dus sorteren met behulp van de vergelijker. Hij/zij mag niet zien welke afbeelding op de kaarten staat. Hij/zij kan kaarten wel schuiven.
* Een leerling is de **teller**. Hij/zij kijkt of het goed gaat allemaal en telt daarnaast het aantal vergelijkingen (dat is het aantal keren dat de vergelijker aangeeft welke kaart lager is)

**Wat te doen**:

Bedenk gezamenlijk een aanpak. Hoe kunnen de kaarten worden gesorteerd? Je kunt eventueel alvast oefenen of het goed gaat en te ontdekken of jullie aanpak werkt.

Als je denkt dat je een goede aanpak hebt, voer je de aanpak uit met alle 9 kaarten. De uitvoerder schudt de kaarten en legt die in een rijtje voor zich neer, met de afbeelding naar beneden. Houd de afbeelding steeds naar beneden, de uitvoerder mag de voorkant van de kaarten niet zien. Vervolgens moet de uitvoerder de kaarten sorteren volgens de gekozen aanpak. De teller telt het aantal keren.

Als het is gelukt, beschrijf dan hieronder in eigen woorden de aanpak. Probeer de aanpak zo precies mogelijk op te schrijven zodat een ander groepje het zou kunnen uitvoeren zonder dat ze vragen hoeven te stellen over de aanpak.

|  |
| --- |
|  |

**Vul in**: voor het sorteren van ....... kaarten hadden we ……… vergelijkingen nodig.

**Werkblad Sorteren van kaarten: bubble sort deel II**

Een bekend algoritme om te sorteren is *bubble sort*. Bij bubble sort loop je de hele rij van kaarten door en vergelijk je steeds de 2 kaarten die naast elkaar liggen. Als ze niet op volgorde liggen, wissel je ze om. Bij 9 kaarten doe je dus 8 vergelijkingen. Dat blijf je herhalen totdat je geen kaarten meer hoeft te verwisselen.





In de onderstaande *stroomdiagram* wordt het algoritme beschreven.



*Ga verder op de achterkant van dit blad.*

Voer het algoritme uit in je groepje en tel het aantal vergelijkingen. **Let op:** zorg dat de uitvoerder de kaarten niet kan zien, leg die dus met de afbeelding naar beneden.

Vul daarna de onderstaande tabel in: wat is het aantal vergelijkingen dat nodig is voor dit algoritme bij 9 kaarten? En hoeveel vergelijkingen komen er bij als je één kaart meer hebt?

|  |  |
| --- | --- |
| **Aantal kaarten** | **Aantal vergelijkingen** |
| 9 |  |
| 10 (dus één kaart extra) |  |

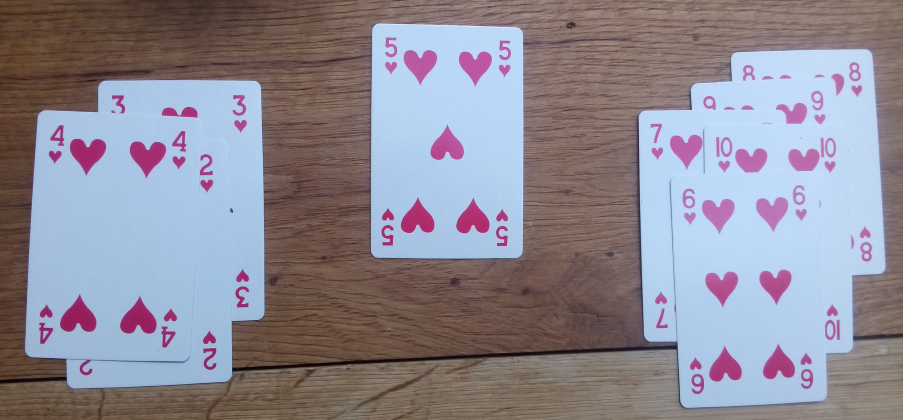
Bedenk een manier om dit algoritme nog wat slimmer te maken. Beschrijf die aanpassing hieronder in je eigen woorden, je hoeft geen nieuw stroomdiagram te maken. Probeer de aanpak weer zo precies mogelijk op te schrijven zodat een ander groepje het zou kunnen uitvoeren zonder dat ze vragen hoeven te stellen over de aanpak.

|  |
| --- |
|  |

**Werkblad Sorteren van kaarten: quick sort deel III**

Een bekend algoritme om te sorteren is quick sort. Bij quick sort kies je willekeurig een kaart, dat is de spilkaart. Die spilkaart vergelijk je met alle andere kaarten (bij 9 kaarten heb je dan dus al 8 vergelijkingen). Op die manier verdeel je de kaarten in twee stapels: een stapel met kaarten die hoger zijn dan de gekozen kaart, en een stapel met kaarten die lager zijn dan de gekozen kaart.

Daarna doe je het zelfde voor elk van die twee stapels: kies een willekeurige kaart en vergelijk die met alle andere kaarten in die stapel. Ga zo door totdat alle kaarten zijn gesorteerd.







Op de achterkant van dit werkblad staat het stroomdiagramvan het algoritme.

Voer het algoritme uit in je groepje en tel het aantal vergelijkingen. **Let op:** zorg dat de uitvoerder de kaarten niet kan zien, leg die dus met de afbeelding naar beneden.

Vul daarna de onderstaande tabel in: wat is het aantal vergelijkingen dat nodig is voor dit algoritme bij 9 kaarten? En hoeveel vergelijkingen komen er bij als je één kaart meer hebt?

|  |  |
| --- | --- |
| **Aantal kaarten** | **Aantal vergelijkingen** |
| 9 |  |
| 10 (dus één kaart extra) |  |



**Werkblad Sorteren van kaarten: insertion sort deel IV**

Een bekend algoritme om te sorteren is *insertion sort*. Bij insertion sort bouw je een nieuwe rij op die gesorteerd is. Je voegt een nieuwe kaart toe op de juiste plek in de nieuwe rij. Je begint dus als volgt: pak de eerste kaart uit de stapel, dat is de eerste kaart in de nieuwe rij. Pak een volgende kaart uit de stapel en voeg die toe aan de nieuwe rij. Je zult de nieuwe kaart dus moeten vergelijken met alle andere kaarten om te weten waar je deze nieuwe kaart moet neerleggen. Doe dat vervolgens voor alle kaarten totdat alle kaarten gesorteerd zijn.





In de onderstaande stroomdiagramwordt het algoritme beschreven.



*Ga verder op de achterkant van dit blad.*

Voer het algoritme uit in je groepje en tel het aantal vergelijkingen. Let op: zorg dat de uitvoerder de kaarten niet kan zien, leg die dus met de afbeelding naar beneden.

Vul daarna de onderstaande tabel in: wat is het aantal vergelijkingen dat nodig is voor dit algoritme bij 9 kaarten? En hoeveel vergelijkingen komen er bij als je één kaart meer hebt?

|  |  |
| --- | --- |
| **Aantal kaarten** | **Aantal vergelijkingen** |
| 9 |  |
| 10 (dus één kaart extra) |  |

Hoe kun je het algoritme zoals staat beschreven in het stroomdiagram nog slimmer maken? Bedenk een manier en beschrijf de aanpassing hieronder in je eigen woorden. Je hoeft geen stroomdiagram te maken. Probeer de aanpak wel weer zo precies mogelijk op te schrijven zodat een ander groepje het zou kunnen uitvoeren zonder dat ze vragen hoeven te stellen over de aanpak.

|  |
| --- |
|  |