

Oefenen met Harry de array[]

Arrays zijn supercool, ze zijn onze meest basic manier om een verzameling van elementen te vertegenwoordigen.

Harry de coder werkt ook graag met arrays en hij heeft enkele goede ideeën die hij graag zou omzetten naar werkelijkheid.

Oefeningen met getallenreeksen

Maak 1 klasse en maak gebruik van private static methodes om de verschillende methodes op te roepen. Zoals de String oefeningen.

- **Controleer of een array van getallen gesorteerd is**

Maak een methode aan met de naam `isSorted`. Deze methode neemt een array van ints als argument en geeft een boolean waarde terug. (returntype is boolean)

De methode gaat controleren of de getallen volgens grootte gesorteerd staan. (vb: 1,6,9,222 is gesorteerd en zou dus true terug geven)

EXTRA: zorg er voor dat je een extra parameter mee kan geven waarmee je aangeeft of de getallen van klein naar groot of van groot naar klein gesorteerd staan.

- **Maak een methode aan die de getallen sorteerd**

Maak een methode aan met de naam `sort`. Deze methode neemt een array van ints als argument en geeft deze zelfde (weliswaar aangepaste) array weer terug. (returntype is `int[]`)

De methode gaat de array sorteren op grootte. (vb: 1,8,6 wordt 1,6,8)

EXTRA: zorg ervoor dat je een extra parameter kan meegeven waarmee je aangeeft of de getallen van klein naar groot of van groot naar klein gesorteerd moeten worden.

EXTRA EXTRA: implementeer een ander sorteeralgoritme(bubblesort of quicksort, zie google voor meer uitleg over deze algoritmes, probeer je programma te schrijven op basis van de algoritmes, niet gewoon code kopiëren)

- **Chopchop**

Maak een methode aan met de naam `chopSearch`. Deze methode neemt een array van ints en

een int als argument en geeft een boolean waarde terug. (returntype is boolean)

De methode gaat controleren of een bepaald getal in een array zit. Je kan dit op verschillende manieren doen. Zo zou je bijvoorbeeld alle waarden van de array kunnen gaan vergelijken met de opgegeven waarde.

EXTRA: Je zou ook (aangeraden) telkens de middelste waarde van je array kunnen vergelijken met de opgegeven waarde, zo kan je controleren of de waarde zich in de bovenste of onderste helft van de array zit, het eerste of tweede kwartaal, etc..

Voorwaarde opdat dit zou werken is wel dat de array gesorteerd moet zijn. Controleer dit dus ook. (zie opgave 1)

Probeer dit op verschillende manieren te implementeren.

Oefeningen met karakter

Maak een klasse CharArrayTools aan die zich bezig zal houden met het uitvoeren van verschillende karakteroperaties.

- Letters filteren

Maak een methode aan met de naam filterAlphabet. Deze methode neemt een array van chars als argument en geeft een array van chars terug. (returntype is char[])

De methode zal uit een array van chars alle niet-alfabet waarden filteren. (vb: 'a', ' ', '?', 'b' wordt 'a', 'b')

De array die wordt teruggegeven zal dus altijd kleiner of even groot zijn dan de opgegeven array.

- Letters sorteren

Maak een methode aan met de naam sort. Deze methode neemt een array van chars als argument en geeft deze zelfde (weliswaar aangepaste) array terug. (returntype is char[])

Deze methode zal de karakters in een array sorteren op alfabetische volgorde.

- Genereer een woordenboek

Maak een methode aan met de naam generateDictionary. Deze methode neemt een int als argument en geeft een array van char arrays terug. (Returntype is char[][])

In deze methode zal je een woordenboek genereren dat alle woorden uit alle talen bevat! (alle talen die het alfabet gebruiken)

Hoe zou dit er uit zien? Alsvolgt:

- woord 1: aaa

- woord 2: aab

...

- woord 17576: zzz

Dit woordenboek bevat alle woorden met 3 letters. Maak de methode zodanig aan dat de gebruiker kan aangeven hoeveel letters de woorden mogen bevatten. (vanaf een bepaalde grootte zal dit niet meer lukken, wat is deze grootte?)

Maak Harry trots