

2014 januari examen

Marina Lens

1. Mondeling

Een bepaald zoek/sort algorithm uitleggen:

- selection sort + lineair/binair zoeken
- insertion sort
- bubble sort + optimisaties
- merge sort
- quick sort

--> Je krijgt een aantal speelkaarten, deze gebruik je om te tonen hoe het sortalgorithm werkt. Dan moet je vragen over de code beantwoorden (code wordt NIET gegeven), en uiteindelijk iets kort zeggen over de complexiteit van dit Algorithm (bvb. C van Bubble Sort = n^2 , waarom? omdat je een for lus binnen een andere for lus hebt - dat is voldoende uitleg voor Mvr. Lens).

2. Schriftelijk

1. Datastructuren

Vraag: Je moet een foto album aanmaken. Een foto album bestaat uit max. 50 mappen, met in elke map max. 250 foto's. Je moet van iedere map de naam bijhouden, bvb. "Barcelona". Je kunt geen 2 of meer mappen hebben met dezelfde naam. Voor elke foto moet je de naam (bvb. "Barcelona Strand 1") bijhouden, en de grootte ervan in KB. Ook hierbij mogen er geen 2 foto's zijn met dezelfde naam.

--> Teken de datastructuren en declareer ze in Javascript.

Mogelijke Oplossing:

```
var namen = new Array(50);
var inhouden = new Array(50);
for (var i = 0; i < inhouden.length; i++){
  .....inhouden[i] = new Array(250);
  .....for(var j = 0; j < inhouden[i].length; j++){
    .....inhouden[i][j] = new Array(2);
  .....}
}
```

```
var mappen = [namen,inhouden]
```

2. Functies

A. Vraag: Schrijf de functie om, gegeven de naam (bvb. "strand") en grootte (bvb. "56" //kb) van een foto en de naam van een map, deze foto toe te voegen aan deze map. Indien deze map niet bestaat moet deze worden aangemaakt. Deze functie moet true of false returnen (true = gelukt, false = niet gelukt). Schrijf alle functionaliteiten hiervoor op.

Mogelijke Strategie?

```
function vindMapIndex(mapNaam){
// gegeven de naam van een map, return index; als deze map niet bestaat, return index -1}
function vindLegeFotoIndex(mapIndex){
// gegeven het index van een map, return de eerste index in de map waar nog geen foto staat;
indien de map vol is (= geen lege indexen), return -1}
function voegMapToe(mapNaam){
//gegeven mapNaam en indien vindMapIndex(mapNaam)==-1 (dus map bestaat nog niet), maak
nieuwe map op eerst volgende lege plaats (gemakkelijkst is om deze lege plaats te vinden door in
de rij van namen te kijken waar eerste "undefined" staat)
//returned index van deze nieuwe map}
function voegFotoToe(mapNaam,fotoNaam,aantalKB){ //hoofdfunctie voor deze vraag
...var toegevoegd = false;
...var mapIndex = vindMapIndex(mapNaam);
...if (mapIndex==-1 /*map bestaat niet*/) mapIndex=voegMapToe(mapNaam); //maakt nieuwe
map en bewaar index
...if(mapIndex!=-1){
.....var fotoIndex = vindLegeFotoIndex(mapIndex);
.....if(fotoIndex!=-1){
.....inhouden[mapIndex][fotoIndex][0] = fotoNaam;
.....inhouden[mapIndex][fotoIndex][1] = aantalKB;}
.....toegevoegd = true; //foto is toegevoegd
.....}
...}
...return toegevoegd;
}
```

B. Vraag: Maak een functie die een rij terug geeft met daarin de naam van iedere map en de totale grootte van ieder van deze mappen. Simpel? Neen, want er is een "catch": De rij moet gesorteerd zijn op grootte van map, van kleinste naar grootste. Dus eerste map in deze rij moet de kleinste zijn, 2de de 2de kleinste, etc.. Maak alle functionaliteiten die hierbij nodig zijn.

Mogelijke strategie?

1. Een functie maken die een nieuwe rij returned met daarin de grootte van map 1, map 2, map 3,

... map 50.

De indexen hiervan moeten hetzelfde zijn als de indexen van de namen van de mappen.

Vb. nieuweRij = [505,550,612,65]

Als je namen rij ["Barcelona","Madrid","Spanje","Strand"] is, dan komt 505 overeen met de map "Barcelona", 550 met "Madrid" etc..

2. Maak een aangepaste sort functie (bvb. bubbleSort) die twee rijen als parameters mee neemt. Dit is eenvoudiger dan het klint. De sort vergelijkt alleen de groottes enz. van de grootte rij, het enig dat je moet bijvoegen is dat wanneer dit algorithm een rij hersorteerd (dus rij1[i] naar rij1[j] beweegt bvb.), dat hij ook de namen rij hersorteerd. Uiteindelijk krijg je de namen hersorteerd volgens de grootte van iedere map (deze grootte rij is dan natuurlijk ook gesorted).

3. Maak de hoofdfunctie voor deze vraag die een 2D rij maakt. Eerste D is de namenrij (die gesorteerd wordt met ons aangepaste sort algorithm) en de tweede D is de rij met groottes van deze mappen (die ook gesorteerd is door ons sort algorithm). LET OP: Deze nieuwe rij moet maar evenlang zijn als het aantal mappen die in gebruik zijn, dus niet alle 50 max. aantal mappen (var aantalMappen = vindLegeMapIndex() //aparte functie die gewoon in namen rij kijkt en de eerste lege index ('undefined') teruggeeft).

C. Vraag: Maak een functie die, gegeven n en m, een rij terug geeft met daarin een collage van $n*m$ foto's. Deze collage bestaat uit willekeurige foto's uit willekeurige mappen. Wel bevat het geen 2 dezelfde foto's. Deze collage is een rij die bestaat uit strings. Een string moet er bvb. als volgt uitzien, "Barcelona\Beach1.jpg". Maak alle functionaliteiten die hiervoor nodig zijn.

Mogelijke strategie?

Deze vraag valt goed mee:

```
function checkString(string,array){
...var stringFound = false;
...for (var i = 0; i < array.length && stringFound == false; i++){
.....if(array[i]==string) stringFound = true;
...}
...return stringFound;
}
function collage(n,m){
...var uitArray = new Array(n*m);
...for (var i = 0; i < uitArray.length; i++){
.....var aantalMappen = (vindLegeMapIndex());
.....var mapIndex = parseInt(Math.random()*aantalMappen);
.....var mapString = namen[mapIndex];
.....var aantalFotos = (vindLegeFotoIndex(mapIndex));
.....var fotoIndex = parseInt(Math.random()*aantalFotos);
.....var fotoString = inhoud[mapIndex][fotoIndex][0]; //0 is waar naam zich bevindt
.....var finalString = mapString + "\" + fotoString + ".jpg"; //"\" ipv "\"?
.....if(checkString(finalString,uitArray)==false){ //zorg dat deze foto niet al in collage staat
.....uitArray[i] = finalString;
.....}
```

```
.....else{ i--; }  
...}  
...return uitArray;  
}
```

Greetje Jongen

Vraag 1a:

Schrijf een functie die een markt genereert, deze markt heeft rijen en in elke rij een identiek aantal kraampjes.

Elk kraampje heeft een categorie, kunst, brol, food. Niet alle plaatsen zijn bezet.

Vraag 1b:

Schrijf een functie die in een string de totale aantallen van alle categorieën teruggeeft, dus bv: "Aantal kraampjes met brol: 22, aantal kraampjes met food: 33, aantal kraampjes met kunst: 25, aantal onbezette kraampjes: 21".

Vraag 2:

De organisator wil dingen fair houden en dus geen kraampjes van dezelfde categorie op de plaatsen N - Z - O - W van een gegeven kraampje, schrijf een functie die dit controleert voor een gegeven categorie en true teruggeeft indien de plaatsing correct is.

Vraag 3:

De koning komt langs op de markt, deze wandeld niet graag en stapt dus van de linkerbovenhoek naar rechts,

dan van rechts naar links op de volgende rij, de volgende links naar rechts en zo verder. Hij wil geen lege/onbezette kraampjes tegenkomen.

Schrijf een functie die een voorgestelde markt (als 2D array) meekrijgt en deze herorganiseert zodat alle lege kraampjes in de laatste rijen staan.

(Wat niet gezegd werd op het examen was dat er dan geen rekening moest gehouden worden met de "Faire" positionering van de kraampjes, dus mochten kraampjes van dezelfde categorie WEL naast elkaar staan)

