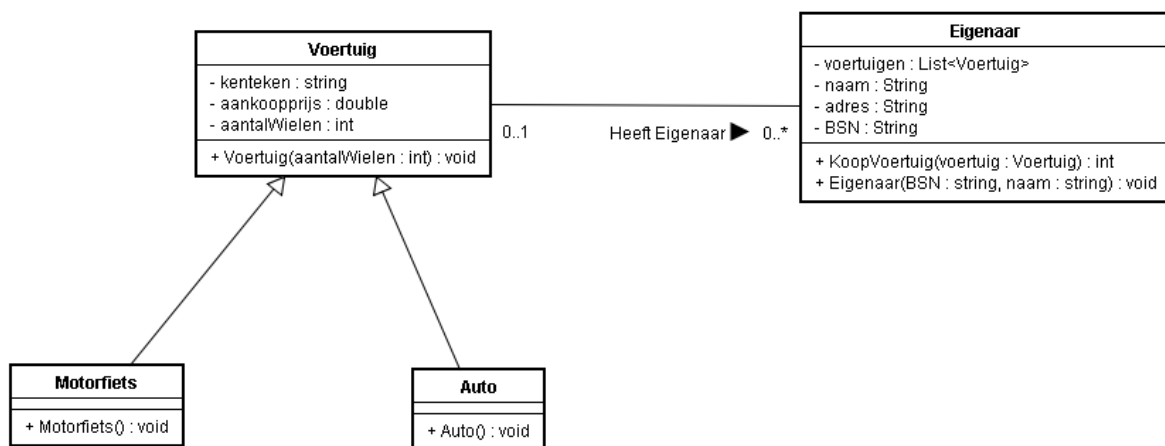


Tentamenvragen Programmeren - Flex Software Engineering 1

Opleiding	HBO ICT - Flex	Toetsvorm	Tentamen
Titel Toetseenheid	Programmeren	Datum	-
Titel Toets	Programmeren	Examinator	Martin Molema
Module	Software Engineering 1	Tijdsduur	2,5 uur
Maximale score	P2 2016/2017 90 punten	Cijfer	(90pt + 10pt) / 10

Toegestane hulpmiddelen	Je mag alleen het boek gebruiken (Programmeren in C#) en geen eigen aantekeningen. Het opzoeken van informatie uit het boek, het lezen en interpreteren hiervan kan veel tijd kosten! Het is natuurlijk niet de bedoeling het boek over te schrijven.
--------------------------------	--



Vraag 1

In bovenstaande klassendiagram worden een aantal klassen beschreven.

Vraag 1a (theorie) (4 + 3 + 3 = 10 punten)

- hoe heten de variabelen binnen een klassendiagram en wat is de standaard zichtbaarheid in C#?
- als je een klasse instantieert met `new Voertuig()` welke speciale functie roep je dan aan?
- Welke relatie geven de pijlen met een open pijlpunt aan in het klassendiagram?

Vraag 1b (5+5+5+5 punten)

Implementeer de klassen uit het bovenstaande klassendiagram: zorg dat alle klassen er zijn en dat de variabelen aanwezig zijn. Maak in C# een *Console Application*.

- Voertuig
 - maak minimaal een constructor die het aantal wielen als parameter heeft.
- auto

- maak een constructor die geen parameters heeft, maar wel meteen het aantal wielen op vier zet
- motor
 - maak een constructor die geen parameters heeft, maar wel meteen het aantal wielen op twee zet
- eigenaar
 - maak een constructor met als parameter de naam en BSN (Burger Service Nummer)

Vraag 2 (4+4+2 = 10 punten)

Voertuigen zijn er natuurlijk om in te rijden, maar dan moet je er wel eerst een kopen.

- implementeer de methode "KoopVoertuig" bij de klasse "Eigenaar". Deze vereist een parameter van de klasse "Voertuig" en voegt deze toe aan de lijst met auto's. Als resultaat geeft deze het aantal voertuigen terug.
- Na het kopen wordt de auto op kenteken gezet. Zorg dat je het kenteken kunt lezen en schrijven bij de klasse 'Voertuig' zonder dat je de variabele voor het kenteken publiekelijk (`public`) schrijfbaar maakt.
- maak één eigenaar en voeg minimaal 3 voertuigen toe van 2 verschillende soorten (auto of motor)

Vraag 3

Gebruik voor het uitvoeren van deze vraag de Visual Studio Solution die je hebt gekregen.

Je kunt je uitwerkingen testen door gebruik te maken van de module NUnit.

Je mag bij het schrijven van deze **functies** niet de volgende ingebouwde functies van de klasse String gebruiken:

- Count()
- Length()

Vraag 3a (4+7 = 11 punten)

Schrijf de onderstaande functie. Je kunt deze testen met de functie 'TestCountLetters'.

- `public static int CountLetters(string tekst)`

Deze functie telt het aantal letters in een woord. Let op: dit zijn dus de letters van A-Z in hoofd- én kleine letters.

Vraag 3b (4+7 = 11 punten)

Je kunt deze testen met de functie 'TestCountLettersRecursive'.

Schrijf een tweede functie die hetzelfde doet, maar dan op een recursieve manier.

- `public static int CountLettersRecursive(string tekst)`

Tip: gebruik de functie `String.Substring(...)` en `String.IsNullOrEmpty(...)`.

Vraag 3c (8 punten)

Implementeer de onderstaande functie. Je kunt deze testen met de functie

`TestFilterOdd();`

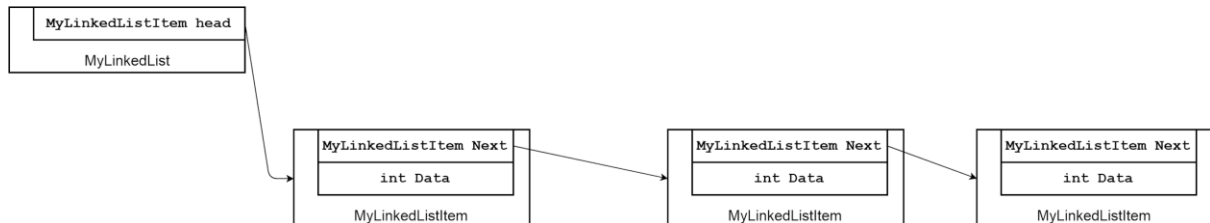
- `public static List<int> FilterOdd(int[] items)`

Deze functie zoekt de on-even getallen uit een opgegeven array, en geeft een nieuwe lijst terug met daarin alleen de gevonden on-even getallen.

Vraag 4

Voor deze vraag wordt een klasse 'MyLinkedList' gebruikt die in de solution van Visual Studio zit. Je moet een aantal opdrachten uitvoeren die getest kunnen worden met NUnit.

De enkelvoudige gelinkte lijst wordt gebruikt om een lijst van integers te kunnen opslaan. Hiervoor wordt een klasse gebruikt met de naam 'MyLinkedListItem'. Dit werkt als volgt:



Vraag 4a (10 punten)

Implementeer de onderstaande functie in de klasse `StudentLinkedList`. Je kunt deze testen met de functie 'TestList1'.

- `public int FindLargest()`

Deze functie zoekt het grootste getal uit de lijst; ga er van uit dat er alleen positieve getallen in de lijst staan.

De functie moet de lijst intact laten.

Vraag 4b (10 punten)

De tweede functie doet precies hetzelfde als bij vraag 4a maar moet geschreven worden door gebruik te maken van recursie. Schrijf de onderstaande functie in de klasse `StudentLinkedList`. Je kunt deze testen met de functie 'TestList2';

- `public int FindLargestRecursive()`

De functie moet de lijst intact laten.

Puntenverdeling en controle

Vraag	Onderdeel	sub	punten	punten per vraag
1	a	1	4	
	a	2	3	
	a	3	3	10
1	b	1	5	
	b	2	5	
	b	3	5	
	b	4	5	20
2	a	1	4	
	a	2	4	
	a	3	2	10
3	a	1	4	
	a	2	7	
	b	1	4	
	b	2	7	
	c	1	8	30
4	a	1	10	
	b	1	10	20

			90	90