Tentamen Verificatie en Testen, 2R460, 28/11/2000, 9.00-12.00

Gebruik van het collegemateriaal over testen is niet toegestaan. Dit tentamen bestaat uit 6 vragen - merk op dat 1-4 over testen samen evenveel punten opleveren als 5 en 6 over verificatie. Veel succes!

1. (a) Het Jelinski-Moranda model is gebaseerd op de exponentiele verdeling. Deze verdeling is geheugenloos, wat wil zeggen dat de in het verleden gevonden fouten geen invloed hebben op kans en frequentie om in de toekomst fouten te vinden. Leg uit waarom de in het Jelinski-Moranda model gebruikte verdeling niet geheugenloos is.

   (b) Het is een probleem om test cases te verzinnen voor sequentiele programma's die een bepaalde coverage garanderen. Leg uit waarom. In de praktijk gebruikt men daarom soms hulpmiddelen die statement coverage registreren voor een tamelijk willekeurig opgestelde verzameling van tests. De tests worden afdoende geacht wanneer bijvoorbeeld 80% van de statements gecovered worden. Noem twee negatieve en één positief aspect van een dergelijke aanpak.

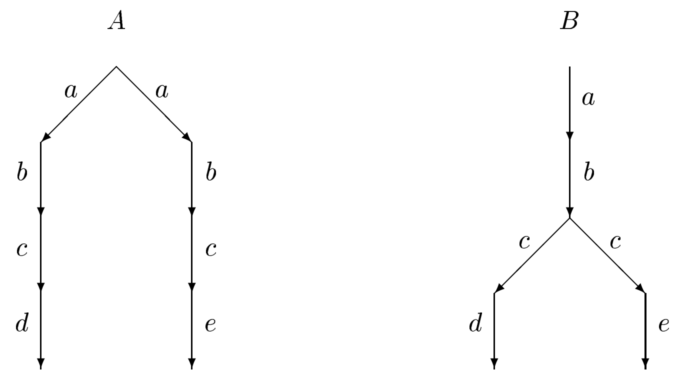
   (c) Geef twee duidelijk verschillende voorbeelden van klassen van mutaties die in een programma zijn aan te brengen ten behoeve van program mutation testing.

   (d) Beschouw de volgende vergelijkingen die een eenvoudige combinatorische schakeling representeren, waarbij a, b en c inputsignalen zijn en e een outputsignaal is.

$$
\begin{aligned}
d &= a \wedge b \\
e &= \neg(d \wedge c)
\end{aligned}
$$

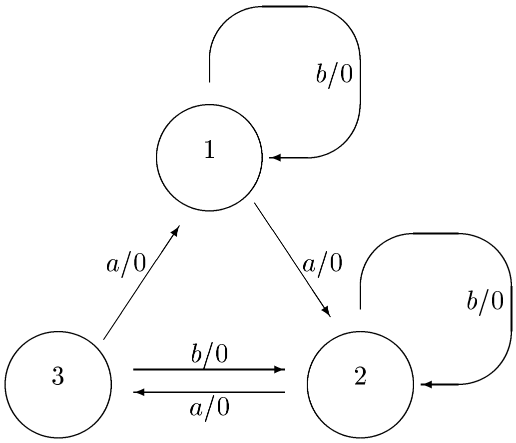
   Geef de inputvectoren waarmee nagegaan kan worden of signaal d getroffen is door een stuck at zero en een stuck at one fault. Geef ook de interpretatie van de waarde van e bij uitvoering van een test met deze inputvectoren.

2. (a) Gegeven zijn de twee onderstaande grafen A en B die processen voorstellen. Beargumenteer dat ze niet te onderscheiden zijn door test scenario's waarbij acties slechts geobserveerd en geblokkeerd kunnen worden.



   (b) Laat zien dat de processen A en B wel van elkaar kunnen worden onderscheiden wanneer de uitvoering van een actie kan worden teruggedraaid (bijvoorbeeld door middel van een undo knop).

3. (a) Onderstaand is gegeven een Mealy machine. Geef een homing sequence, mits die bestaat. Geef in dat geval aan hoe de output van een test met deze sequence moet worden geinterpreteerd. Als de homing sequence niet bestaat, leg dan uit waarom.



(b) Is het bepalen of een synchronizing sequence voor een eindige Mealy machine bestaat berekenbaar? Geldt dit ook voor het opstellen van een synchronizing sequence indien hij bestaat?

4. (a) Wat is een Implementation Conformance Statement en hoe verschilt zij van een ICS proforma.

(b) Stelt u zichzelf voor in de rol van de ingenieur verantwoordelijk voor het testen van de software in een automobiel. Stel dat u een keuze moet maken uit:

i. testen op basis van de theory van testgeneratie op basis van Mealy machines.

ii. testen op basis van statistische analyse van de aard en het moment van opgetreden problemen in de software.

iii. testen op basis van het traject ICS, ICS proforma, testdoelen, test templates, test executies en test verdicts.

Geef uw beargumenteerde keuze, waarbij argumenten betrekking moeten hebben op alle drie mogelijkheden. Beschrijf ook de risico's van uw keuze en schets of en hoe die afgedekt zouden kunnen worden.

5. Beschouw het volgende mutual exclusion algoritme.

   (a) Geef (dwz teken) de Street automaat A die alle oneindige executies van deze processen accepteert (dwz met lege verzameling acceptatiecondities).
   (b) Geef, in termen van $enabled(t)$ en $taken(t)$, een algemene voortgangsconditie in LTL die zorgt voor Eventual Access tot elk van de c_i locaties ($i = 1, 2, 3$).
   (c) Geef de Street automaat A_{EA} door bij automaat A een verzameling ACC van acceptatiecondities te geven die zorgen voor Eventual Access tot elk van de c_i locaties ($i = 1, 2, 3$).
   (d) Geef ook de (minder eisende) verzameling ACC' van acceptatiecondities die zorgen voor Eventual Access tot c_1 en tot minstens één, maar niet noodzakelijk beide, van de locaties $\{c_2, c_3\}$.
   (e) Eventual Access tot c_1 en c_2 wordt weergegeven door de LTL formule $\varphi = Fc_1 \wedge Fc_2$. Geef, eventueel maar niet noodzakelijk via de standaard constructie, een Street automaat $A_{\neg\varphi}$ voor de negatie van φ.
   (f) Geef de product Street automaat $A_{EA} \times A_{\neg\varphi}$. (Voor het geval van de eerstgevraagede acceptatie condities, ACC.)
   (g) Beargumenteer dat $A_{EA} \times A_{\neg\varphi}$ geen paden heeft die aan de acceptatie condities van $A_{EA} \times A_{\neg\varphi}$ voldoen.

6. Beschouw nogmaals de twee grafen uit opgave 2. Laat alle acties op de transities doorschuiven naar de knoop erna en beschouw ze als proposities die waar zijn aesa ze in een knoop vermeld staan (in de startknopen geldt nu dus $\neg a \wedge \neg b \wedge \neg c \wedge \neg d \wedge \neg e$).

   (a) Geef, als die er is, een modale logica formule die, geevalueerd in de begintoestand, onderscheid maakt tussen graaf A en graaf B. Dat wil zeggen dat de formule waar is voor één van de grafen, maar niet voor de andere; geef aan voor welke hij waar is. Als zo'n formule er niet is, beargumenteer dan waarom niet.
   (b) Geef, als die er is, een temporele logica formule die, geevalueerd in de begintoestand, onderscheid maakt tussen graaf A en graaf B. Dat wil zeggen dat de formule waar is voor één van de grafen, maar niet voor de andere; geef aan voor welke hij waar is. Als zo'n formule er niet is, beargumenteer dan waarom niet.

1a:	2	2a:	5	3a:	5	4a:	5	5a:	5	6:	a:5
1b:	3	2b:	5	3b:	4	4b:	10	5b:	5		b:5
1c:	3							5c:	5		
1d:	3							5d:	5		
								5e:	5		
								5f:	5		
								5g:	5		

Cijfer: (totaal aantal punten + 10)/10