



Tentamen Software Engineering
9 januari 1998

1. Je bent als consultant ingehuurd door Fast Shoes, een bedrijf dat schoenen produceert en verkoopt. De directeur van Fast Shoes wil dat jouw organisatie een advies uitbrengt over de te gebruiken strategie bij de ontwikkeling van een systeem dat de voorraad van Fast Shoes bijhoudt, te beginnen bij de inkoop van leer, via de productie van schoenen, de distributie naar de winkels, tot en met de verkoop van de schoenen. Bij eerdere ontwikkelprojecten heeft Fast Shoes steeds de watervalmethode gehanteerd. Schrijf een rapportje voor de directeur van Fast Shoes waarin je beargumenteert om in dit geval een iteratieve ontwikkelstrategie te kiezen, zoals RAD, prototyping of evolutionair ontwikkelen.

Denk erom dat managers niet graag een memo van meer dan een halve pagina lezen.

2. In hedendaagse methoden voor object-georiënteerde analyse en ontwerp komen bijna altijd de volgende soorten diagrammen voor: class diagrams, state diagrams en sequence diagrams. Geef kort aan wat in deze diagrammen wordt weergegeven, en geef een niet-triviaal voorbeeldje van elk.
3. Je moet software ontwikkelen die controleert of een bankafschrift correct is. De gegevens die je hierbij nodig hebt komen uit 3 verschillende bestanden:
- het saldo aan het begin en het eind van de maand,
 - het nummer, de datum en het bedrag van elke uitgeschreven cheque, en
 - de datum en het bedrag van elke storting.
- a. Geef een definitie van wat "correct" hier zou kunnen betekenen, en geef een daaraan aangepaste versie van bovengenoemde eis.
- b. Teken, uitgaande van de in onderdeel a) nauwkeurig geformuleerde eis, een dataflow diagram voor dit stukje software. Verklaar hierbij de symbolen die je gebruikt.
4. Gegeven is het volgende stukje programma:

```
i:= 1; pos:= 0;
while i ≤ n do
    if value[i] > 0 then pos:= pos + 1 endif;
    i:= i + 1
enddo;
if pos > 0 then
    procpos:= n/pos;
    print("percentage positieve scores =", procpos)
else print("geen positieve scores")
endif;
```

- a) Teken de flowgraph voor dit programmafragment.
 - b) Geef een of meer testgevallen (waarden voor n en value[1 .. n]) zodanig dat er een 100% statement-overdekking (All-Nodes coverage) is.
 - c) Geef een of meer testgevallen zodanig dat er een 100% segment-overdekking (All-Edges coverage) is.
 - d) Wat is de cyclomatische complexiteit van dit stukje programma? Verklaar je antwoord.
5. Geef aan of en hoe je de volgende metrieken kunt gebruiken voor het bepalen van de benodigde onderhoudsinspanning:
- a) de lengte van het programma
 - b) het aantal keuze-instructies
 - c) het aantal globale variabelen in het programma
 - d) de gemiddelde afstand tussen de declaratie van een variabele en het gebruik van die variabele

Maak bij het beantwoorden onderscheid tussen correctief, adaptief en perfectief onderhoud. Verklaar je antwoord.

Normering:

1	2	3	4	5
15	15	20	20	20

Cijfer: 1 + aantal behaalde punten/10