

Toetsing van programmatuur

- Doel is om foute te vind—nie om te wys die programmatuur is “korrek” nie.
- Toetsing neem tyd en kos geld; maak dit *effektief* (kry baie foute met min toetse)
- Ontwerp toetsgevalle—toevoer data wat bedoel is om spesifieke eienskappe te toets:
 - wat gebeur as die “then”-deel van 'n “if”-opdrag uitvoer?
 - wat gebeur as 'n while-lus glad nie uitvoer nie?
 - wat gebeur as die toevoerdata buite die spesifikasie se grense is? (byvoorbeeld 'n negatiewe getal as die spesifikasie 'n positiewe getal vereis)

1

Kan jy die volgende riglyne motiveer?

- Toetsgevalle moet die korrekte afvoer beskryf vir elke toevoer.
- Programmeerders moenie hulle eie kode toets nie.
- Sluit geldige sowel as ongeldige toetsdata in.
- Dokumenteer toetsgevalle sodat toetse herhaal kan word as 'n stelsel verander word.
- Aanvaar dat foute gevind sal word.
- As baie foute gevind word in 'n sekere komponent, is daar waarskynlik nog baie meer foute in daardie komponent.

2

Kodeinspeksie

- Verduidelik wat jou kode doen aan iemand anders.
- Baie foute word in 'n beperkte tyd op dié manier gevind.
- Tipiese strategie vir kodeinspeksie:
 - Die programmeerder wat die kode geskryf het verduidelik hoe alles werk aan 'n ander programmeerder wat kennis het van die komponent en programmeertaal.
 - Die programmeerder wat die kode geskryf het moet self beter oplossings uitdink en foute regmaak; foute word gedokumenteer.

3

Ontwerp van toetsgevalle

'n Program aanvaar drie positiewe heel getalle wat die lengtes van die sye van 'n driehoek voorstel. Die program se afvoer is een van die volgende: (1) gelyksydige driehoek (2) gelykbenige driehoek (3) reghoekige driehoek (4) ongelyksydige driehoek.

Ontwerp 'n stel toetsgevalle om die program deeglik te toets.

4

(1) Funksionele toetsing

- Funksionele toetsing: gebruik die *spesifikasie* om toetse te ontwerp.
- Toets elke prosedure as 'n geheel: toetsgevalle bestaan uit geldige en ongeldige toevoerwaardes en ooreenstemmende resultate.
- Toegang tot bronkode onnodig.

5

(a) Toetsing van grenswaardes

- Vir elke parameter kies VIER grenswaardes (minimum waarde, waarde net bo minimum, maksimum waarde en 'n waarde net onder maksimum) en EEN tussenin waarde
- Kies alle kombinasies van 'n grenswaarde vir EEN parameter saam met tussenin waardes vir die ander parameters. (Motivering: min foute is afhanklik daarvan dat *meer as een* parameter by 'n grenswaarde moet wees.)
- $4n + 1$ toetsgevalle nodig vir prosedure met n parameters.

6

Wanneer werk toetse vir grenswaardes goed?

- Prosedures met parameters wat onafhanklik is van mekaar.
- Parameters kan ten minste 5 verskillende waardes aanneem.
- Toevoerwaardes word gebruik in relatief ingewikkelde berekenings of as indekse. Dit beteken die waarskynlikheid van foute is redelik hoog.

7

Oefening: grenswaardes

Ontwerp toetse vir 'n prosedure wat driehoekige klassifiseer. Aanvaar die minimum lengte van 'n sy is 1 en die maksimum 200.

- Daar is 3 toevoerwaardes (3 lengtes)
- Daar moet dus 13 toetse wees.
- Toevoer waardes: 1, 2, 100, 199, 200
- Lys alle toetsgevalle en die verwagte resultate.
- Is die toetse sinvol?

8

Oefening: grenswaardes

Ontwerp toetse vir 'n roetine wat 'n datum in die formaat dd/mm/yy omskakel na die formaat ddd/yy (Juliaanse datum formaat)

- Daar is 3 toevoer parameters (13 toetsgevalle)
- Bepaal die grenswaardes
- Is die toetse sinvol?

9

(b) Toetsing van robuustheid

- Kies waardes *weerskante* van elke grens—SEWE waardes dus.
- Twee waardes is dus ongeldig.
- Tipiese foute wat gevind word:
 - Indeks buite grense (veral nuttig om C programme te toets)
 - Deel deur nul
 - Poging om nul-wyser te gebruik
 - Oorvloei of ondervloei van buffers

10

(c) Toetsing van ekstreme kondisies

- Toets wat gebeur as *meer as een* parameter 'n grenswaarde aanneem.
- Aantal toetsgevalle: 5^n (5 waardes, alle kombinasies)
- Baie meer moeite, deegliker.
- Gebruik wanneer parameters afhanklik is van mekaar en betroubaarheid baie belangrik is.

11

Toetsing van spesiale waardes

- Soms weet ons dat sekere toevoerwaardes "spesiaal" is en kan dan spesifiek toetse vir sulke gevalle insluit.
- Dikwels baie effektief, alhoewel geen riglyne anders as kennis van die aard van 'n procedure.
- Voorbeelde:
 - Vir kode wat datums hanteer is Februarie spesiaal omdat dit verband hou met skrikkeljare.
 - Die waarde 0 is spesiaal saam met deling
 - Leë karakterstringe en leë leërs is bekende gevalle wat probleme veroorsaak.

12

Verdeling van toevoerdata in ekwivalensieklasse

- Verdeel toevoerdata in onderling uitsluitende klasse wat *eenders* is op een of ander manier en selekteer verteenwoordigende toetsdata uit elke klas.
- Voorbeeld: parameter se waarde is negatief, 0 of positief; kies byvoorbeeld -3, 0 en 5.
- Idee agter die tegniek: waardes uit dieselfde klas word dikwels op dieselfde manier hanteer deur 'n program.
- Die sukses van die tegniek hang af van geskikte definisies van die klasse

13

Verskillende tipes ekwivalensieklas toetsing

- Swak ekwivalensieklas toetsing: verteenwoordig elke ekwivalensieklas minstens EEN keer in die toetsdata.
- Sterk ekwivalensieklas toetsing: gebruik alle moontlike kombinasies van waardes uit alle ekwivalensieklasse.

14

Voorbeeld

'n Prosedure het 3 heeltallige parameters a , b en c . Verdeel a en b in die klasse negatief, 0 en positief en c in die klasse ewe en onewe.

1. Ontwerp geskikte toetsdata volgens die swak ekwivalensieklas tegniek.
2. Hoeveel toetsgevalle is nodig vir sterk ekwivalensieklas toetsing?

15

Toetsing volgens beslissingstabelle

- Tegniek nuttig vir programme wat komplekse logika bevat en verskillende resultate gee onder verskillende omstandighede.
- Lys alle kondisies en ooreenstemmende resultate.
- Selekteer toevoerwaardes om verskillende kondisies waar te maak en kyk of die verwagte resultaat gelewer word.

16

Voorbeeld

Toetsdata vir program wat driehoeke klassifiseer.

Toetsgeval	1	2	3	4	5	6	7	8	9	10	11
$a < b + c$	F	T	T	T	T	T	T	T	T	T	T
$b < a + c$	*	F	T	T	T	T	T	T	T	T	T
$c < a + b$	*	*	F	T	T	T	T	T	T	T	T
$a = b$	*	*	*	T	T	T	T	F	F	F	F
$a = c$	*	*	*	T	T	F	F	T	T	F	F
$b = c$	*	*	*	T	F	T	F	T	F	T	F
Geen driehoek	x	x	x								
Ongelyksydig											x
Gelykbenig							x		x	x	
Gelyksydig				x							
Onmoontlik					x	x		x			

Selekteer toetsdata deur waardes te kies wat aan die vereistes van die tabel voldoen. (Vereistes aangedui as * kan enige waardes aanneem.)