CSCE 747 Software Testing and Quality Assurance

Lecture 05 –

Wrapping Up Functional Testing

Last Time

- Decision Table Based Testing
- Ch 7 pp 103-116
- Decision Trees
- Decision Tables for Business Logic
- Decision Tables for Testing
- Junit testing

Today

- Wrapup Functional Testing
- Ch 8 pp 117-127
- Testing Effort
- Testing Efficiency
- Testing Effectiveness
- Guidelines
- Case Study – Insurance Premium

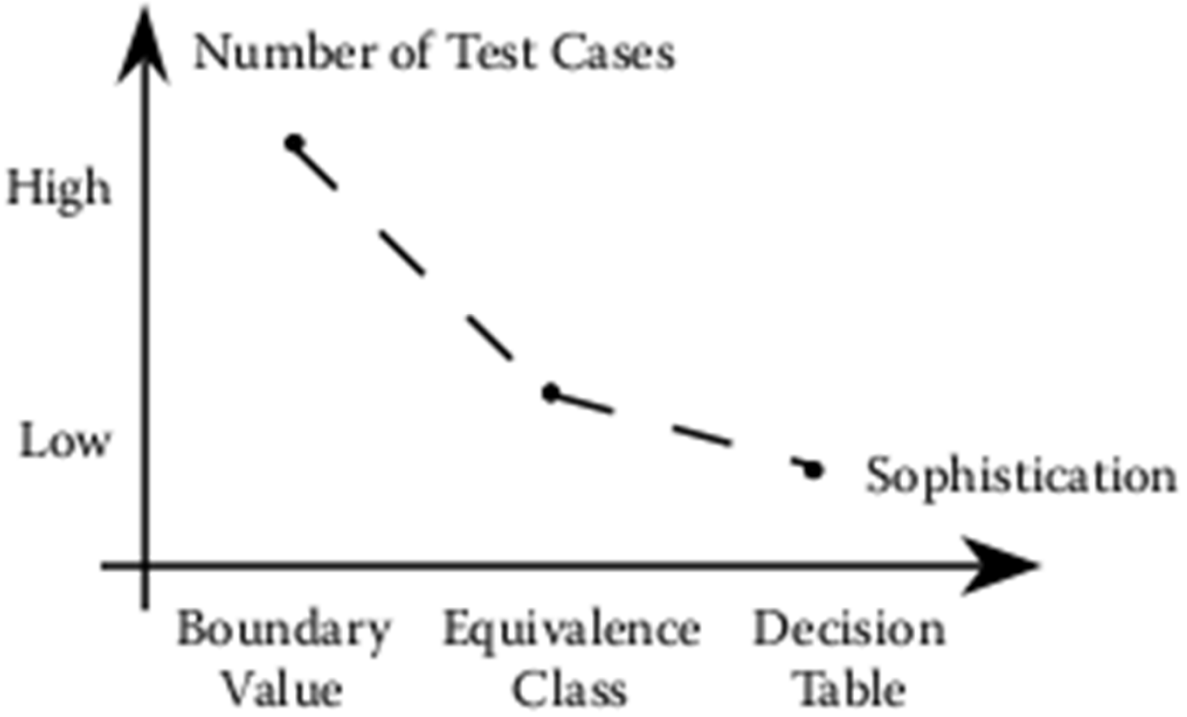
Functional Testing Review

- Black-box testing –
- Approaches
  - Boundary values
  - Equivalence Class
  - Decision Table

Testing Effort

Testing Effort

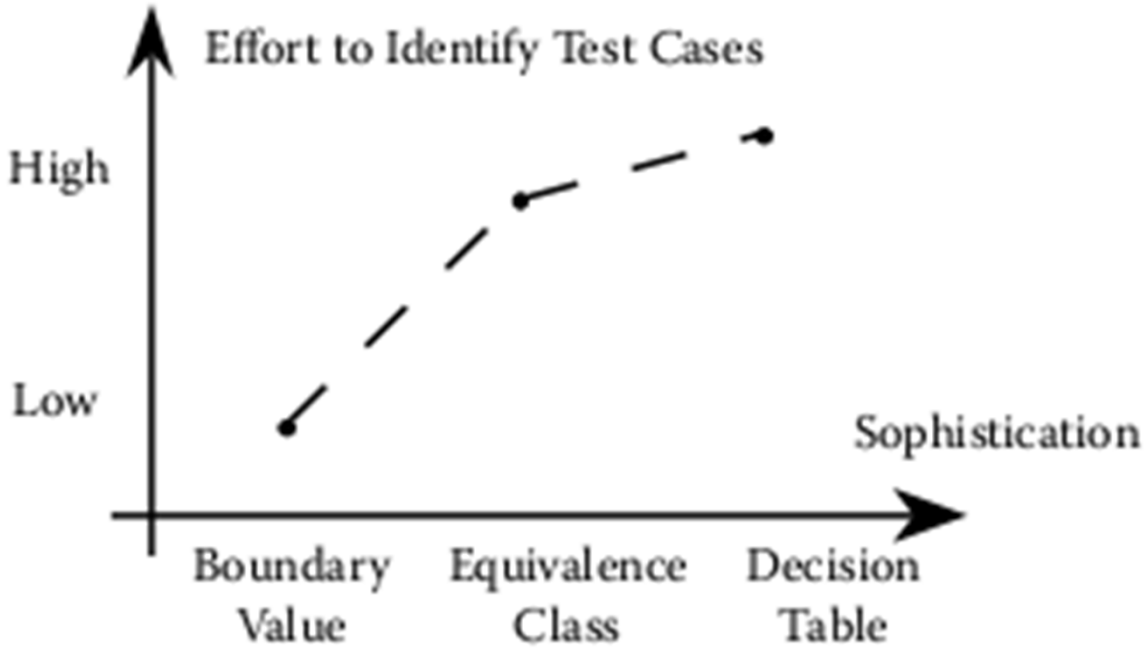
- **Fig 8.1 Trendline of test cases by testing method**



Jorgensen, Paul C. Software Testing
A Craftsman Approach

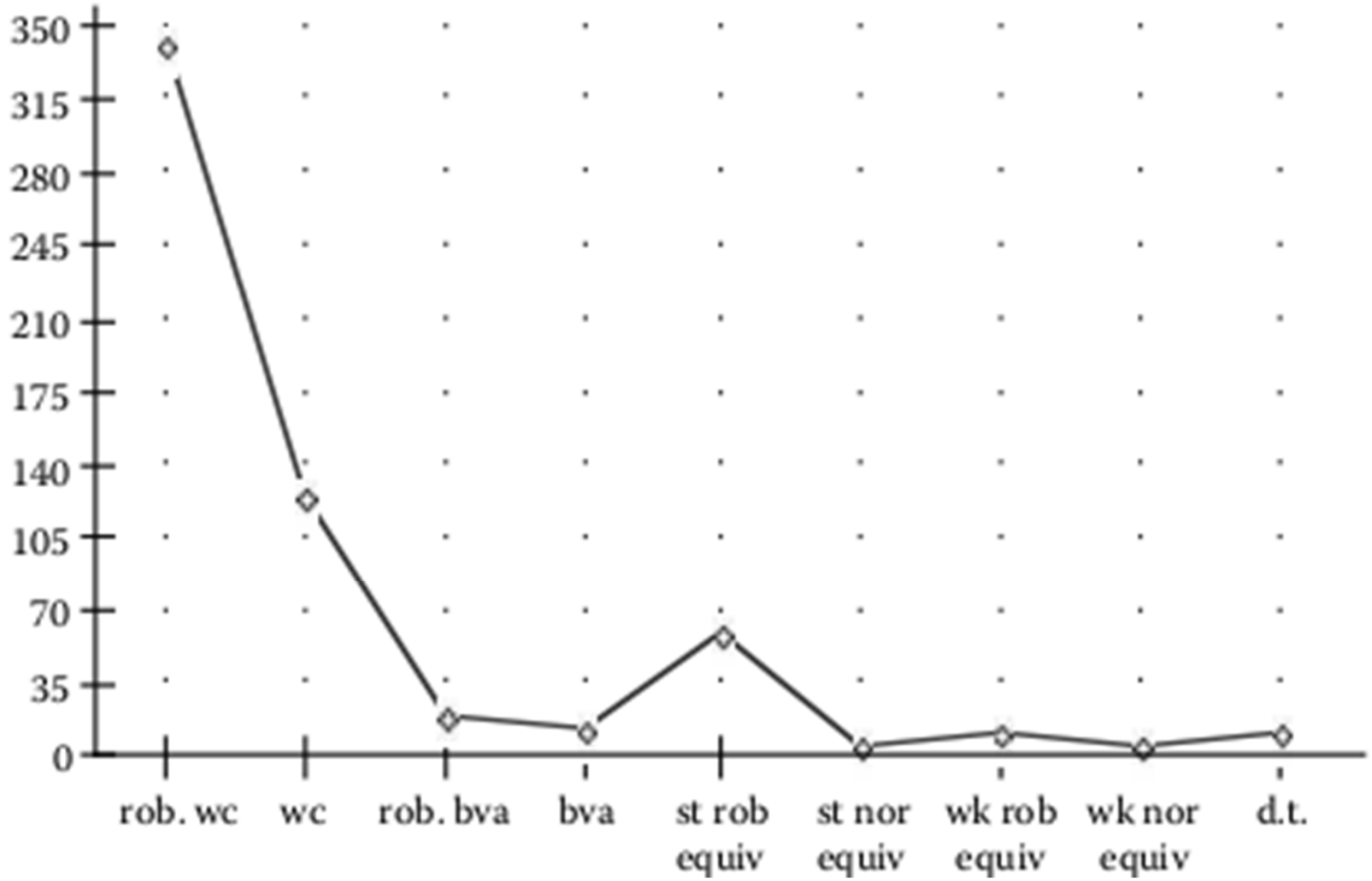
Testing Effort

- **Fig 8.1 Trendline of test case identification by testing method**
-



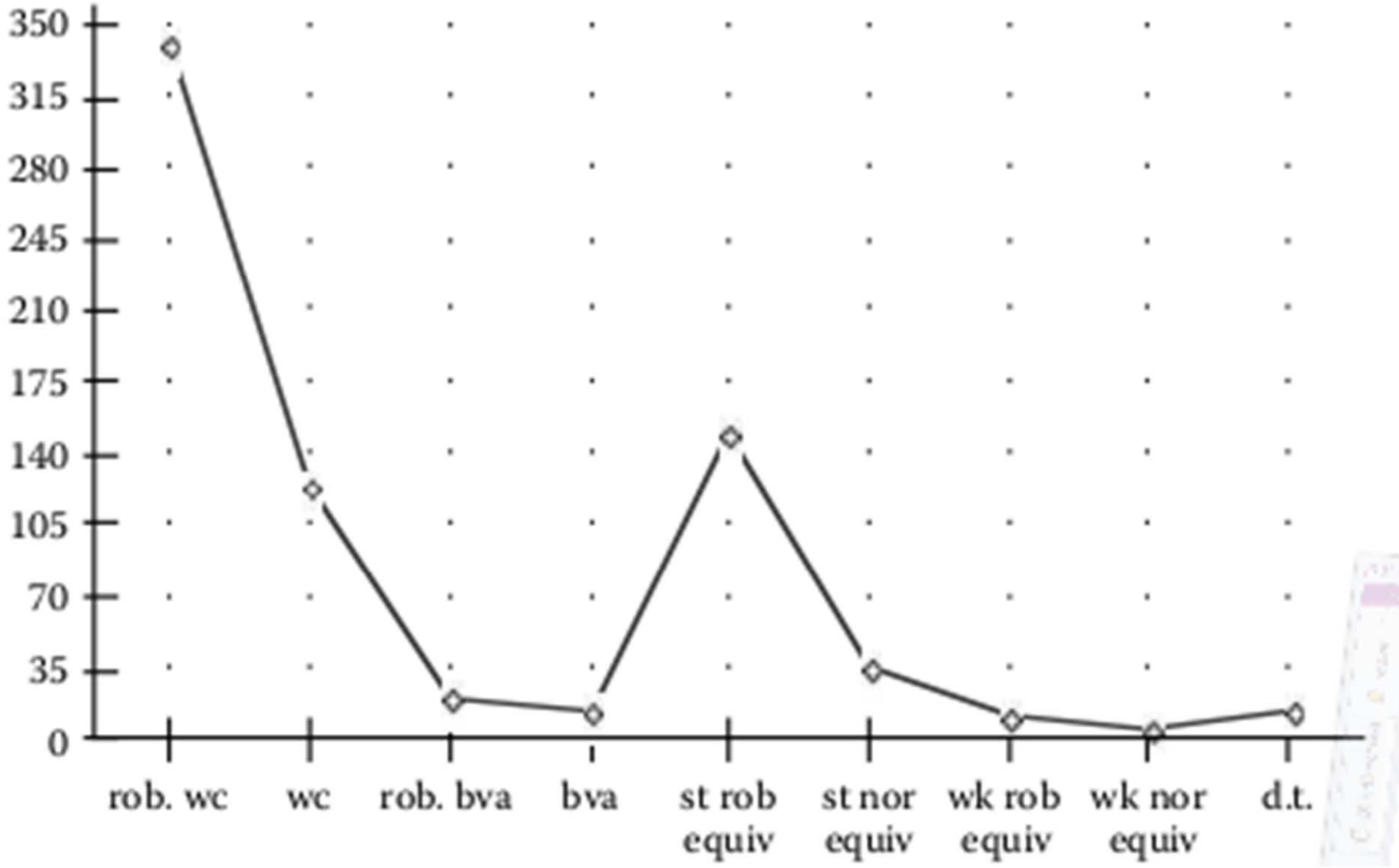
Jorgensen, Paul C. Software Testing
A Craftsman Approach

Test Cases per Method –Triangle Problem



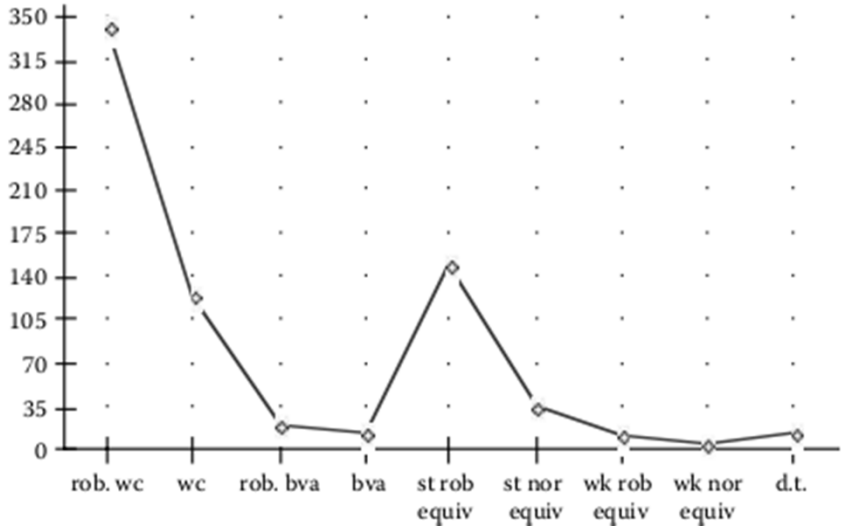
Jorgensen, Paul C. Software Testing
A Craftsman Approach

Test Cases per Method –NextDate



Jorgensen, Paul C. Software Testing
A Craftsman Approach

Test Cases per Method –Commission Problem



Jorgensen, Paul C. Software Testing
A Craftsman Approach

Limitation of Functional Testing

- fundamental limitation of functional testing:
  - the twin possibilities of
  - gaps of untested functionality and
  - redundant tests.

Guidelines

Looking for your keys Story

- Here is one of my favorite testing stories. An inebriated man was crawling around on the sidewalk beneath a streetlight. When a policeman asked him what he was doing, he replied that he was looking for his car keys. "Did you lose them here?" the policeman asked. "No, I lost them in the parking lot, but the light is better here."

Analogy to Testing

- **This little story contains an important message for testers:**
  - **Testing for faults that are not likely to be present is pointless.**
  - **It is far more effective to have a good idea of the kinds of faults that are most likely (or most damaging) and then to select testing methods that are likely to reveal these faults.**

Attributes for Selecting Testing Methods

- Whether the variables represent physical or logical quantities
- Whether dependencies exist among the variables
- Whether single or multiple faults are assumed
- Whether exception handling is prominent

an "Expert System" on Functional Testing Approach Selection

1. "If the variables refer to physical quantities, domain testing and equivalence class testing are indicated.
2. If the variables are independent, domain testing and equivalence class testing are indicated.
3. If the variables are dependent, decision table testing is indicated.
4. If the single fault assumption is warranted, boundary value analysis and robustness testing are indicated.
5. If the multiple fault assumption is warranted, worst-case testing, robust worst-case testing, and decision table testing are indicated.
6. If the program contains significant exception handling, robustness testing and decision table testing are indicated.
7. If the variables refer to logical quantities, equivalence class testing and decision table testing are indicated."

Decision Table for Technique Selection

Table 8.1 Appropriate Choices for Functional Testing

c1	Variables (P, physical; L, logical)	P	P	P	P	P	L	L	L	L	L
c2	Independent variables?	Y	Y	Y	Y	N	Y	Y	Y	Y	N
c3	Single fault assumption?	Y	Y	N	N	—	Y	Y	N	N	—
c4	Exception handling?	Y	N	Y	N	—	Y	N	Y	N	—
a1	Boundary value analysis		x								
a2	Robustness testing	x									
a3	Worst-case testing				x						
a4	Robust worst case			x							
a5	Weak robust equivalence class	x		x			x		x		
a6	Weak normal equivalence class	x	x				x	x			
a7	Strong normal equivalence class			x	x	x			x	x	x
a8	Decision table					x					x

Case Study From Text

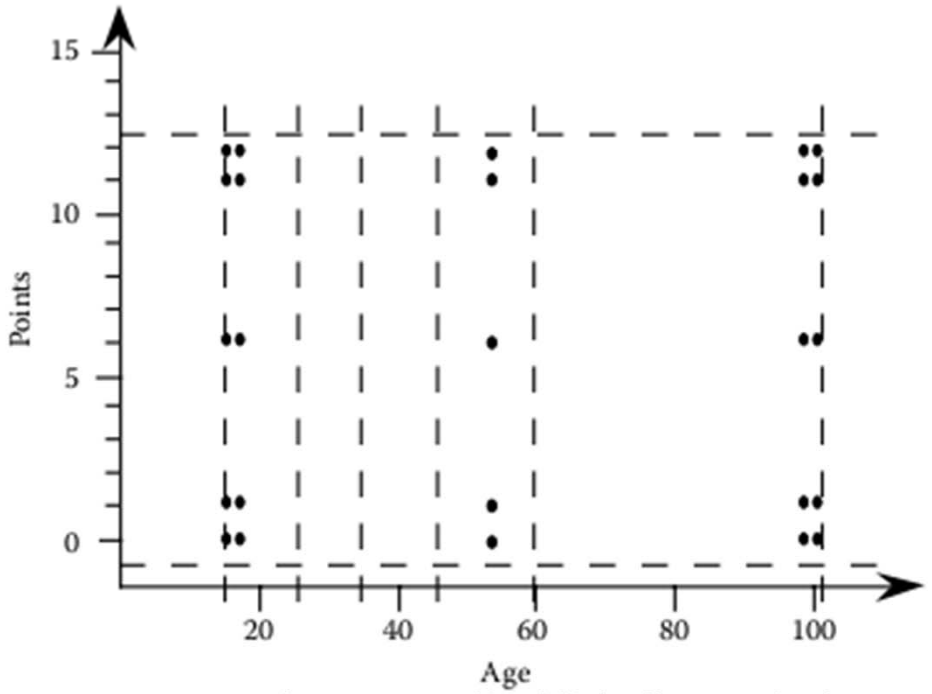
- "An insurance premium program computes the semiannual car insurance premium based on two parameters:
  - the policyholder's age and
  - driving record:
- Using Premium = BaseRate*ageMultiplier - safeDrivingReduction
- The ageMultiplier is a function of the policyholder's age, and
- the safe driving reduction is given when the current points (assigned by traffic courts for moving violations) on the policyholder's driver's license are below an age-related cutoff.
- Policies are written for drivers in the age range of 16 to 100.
- Once a policyholder has 12 points, the driver's license is suspended (thus, no insurance is needed).
- The BaseRate changes from time to time; for this example, it is $500 for a semiannual premium."

Safe Driving Reduction Table

Age Range	*Age Multiplier*	*Points Cutoff*	*Safe Driving Reduction*
$16 \leq age < 25$	2.8	1	50
$25 \leq age < 35$	1.8	3	50
$35 \leq age < 45$	1.0	5	100
$45 \leq age < 60$	0.8	7	150
$60 \leq age < 100$	1.5	5	200

Worst–case boundary value test cases

Variable	*Min*	*Min+*	*Nom.*	*Max–*	*Max*
Age	16	17	54	99	100
Points	0	1	6	11	12



Jorgensen, Paul C. Software Testing
A Craftsman Approach

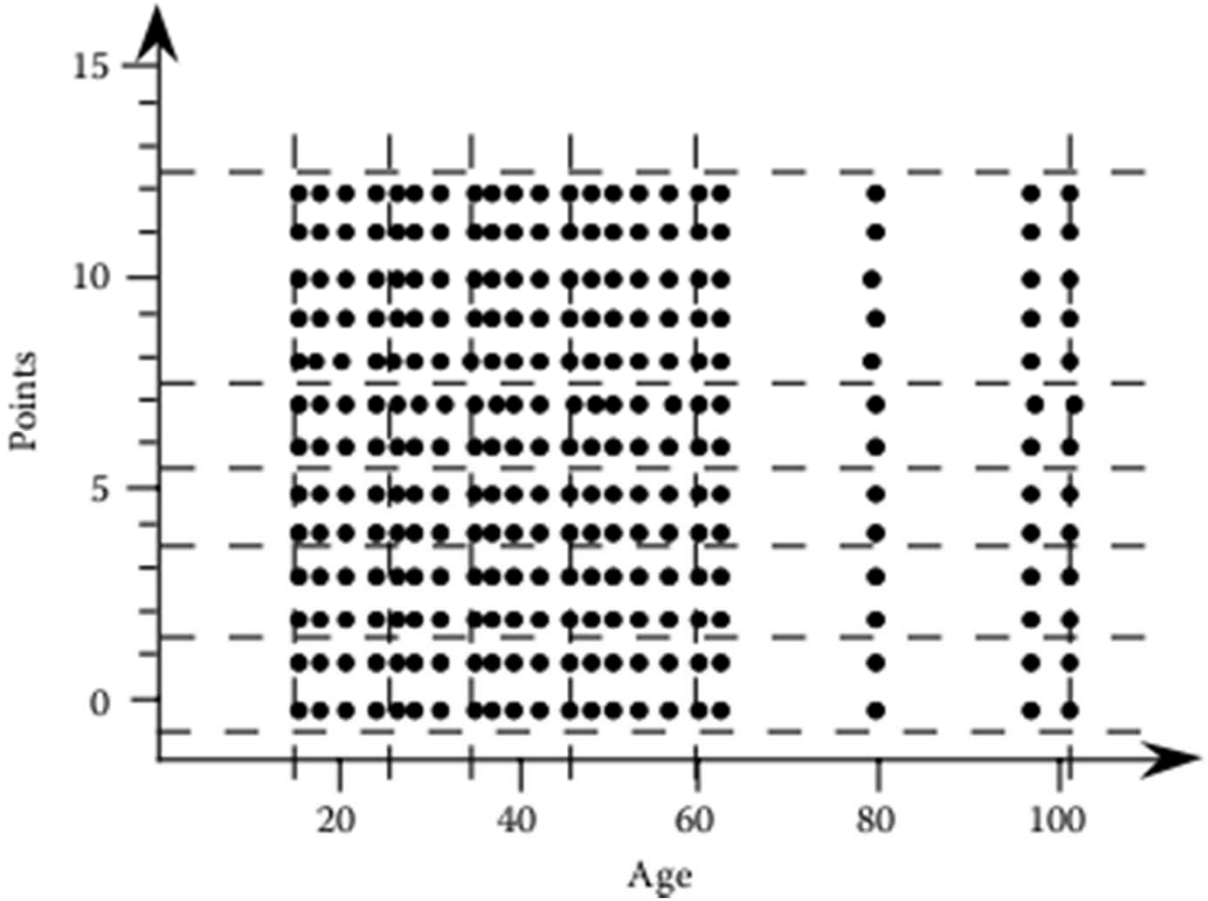
Refinement of Partition

- A1 = {age: 16 ≤ age < 25}
- A2 = {age: 25 ≤ age < 35}
- A3 = {age: 35 ≤ age < 45}
- A4 = {age: 45 ≤ age < 60}
- A5 = {age: 60 ≤ age < 100}
- P1 = {points = 0, 1}
- P2 = {points = 2, 3}
- P3 = {points = 4, 5}
- P4 = {points = 6, 7}
- P5 = {points = 8, 9, 10, 11, 12}
- A x P has 25 induced equivalence classes

Table 8.2 Detailed Worst-Case Values

Variable	*Min*	*Min+*	*Nom.*	*Max–*	*Max*
Age	16	17	20	24	
Age	25	26	30	34	
Age	35	36	40	44	
Age	45	46	53	59	
Age	60	61	75	99	100
Points	0	n/a	n/a	n/a	1
Points	2	n/a	n/a	n/a	3
Points	4	n/a	n/a	n/a	5
Points	6	n/a	n/a	n/a	7
Points	8	9	10	11	12

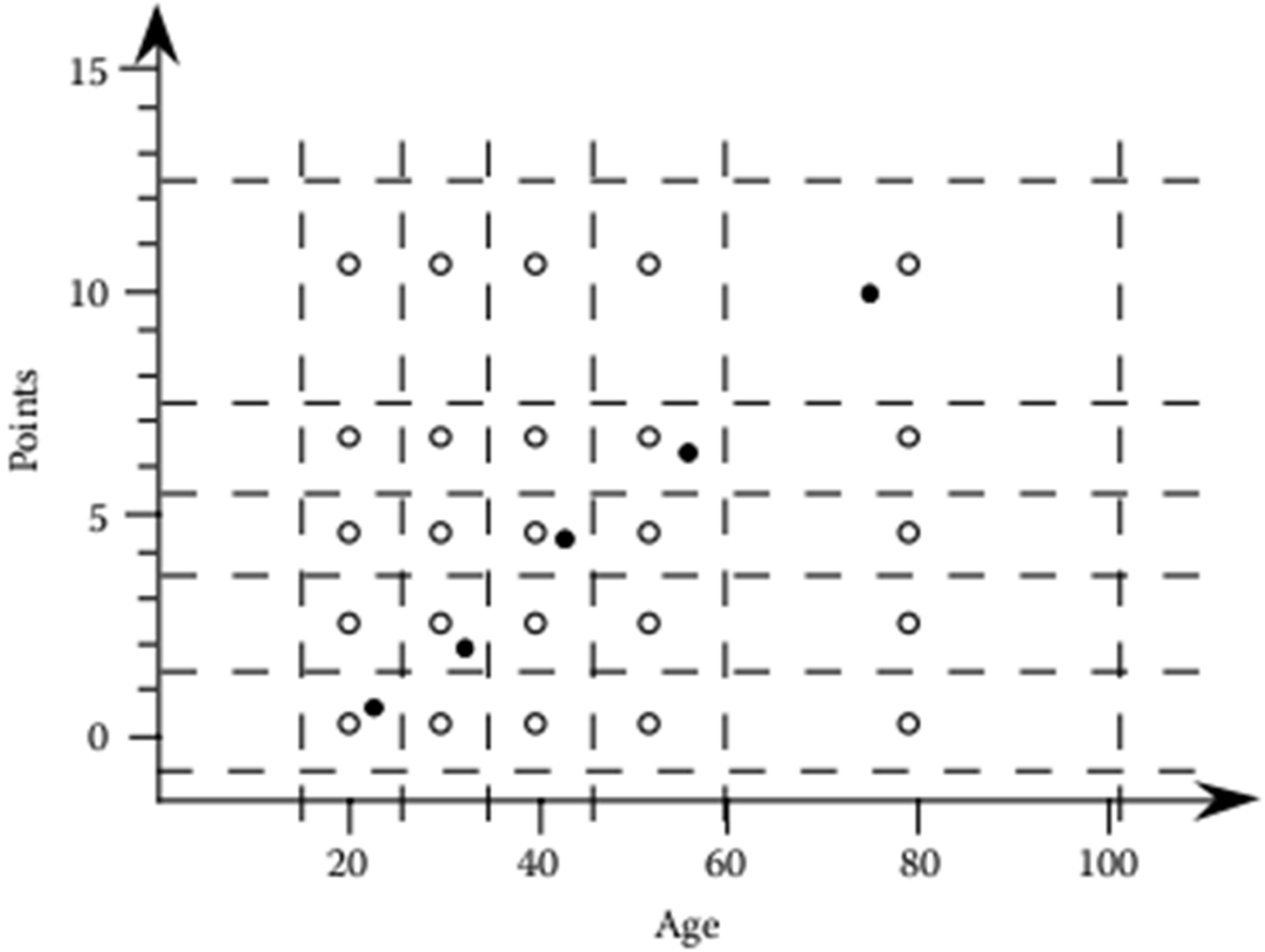
Fig 8.8 Detailed worst-case boundary test cases



Jorgensen, Paul C. Software Testing
A Craftsman Approach

Weak and Strong Eq. Class Test Cases

- Weak
- Strong



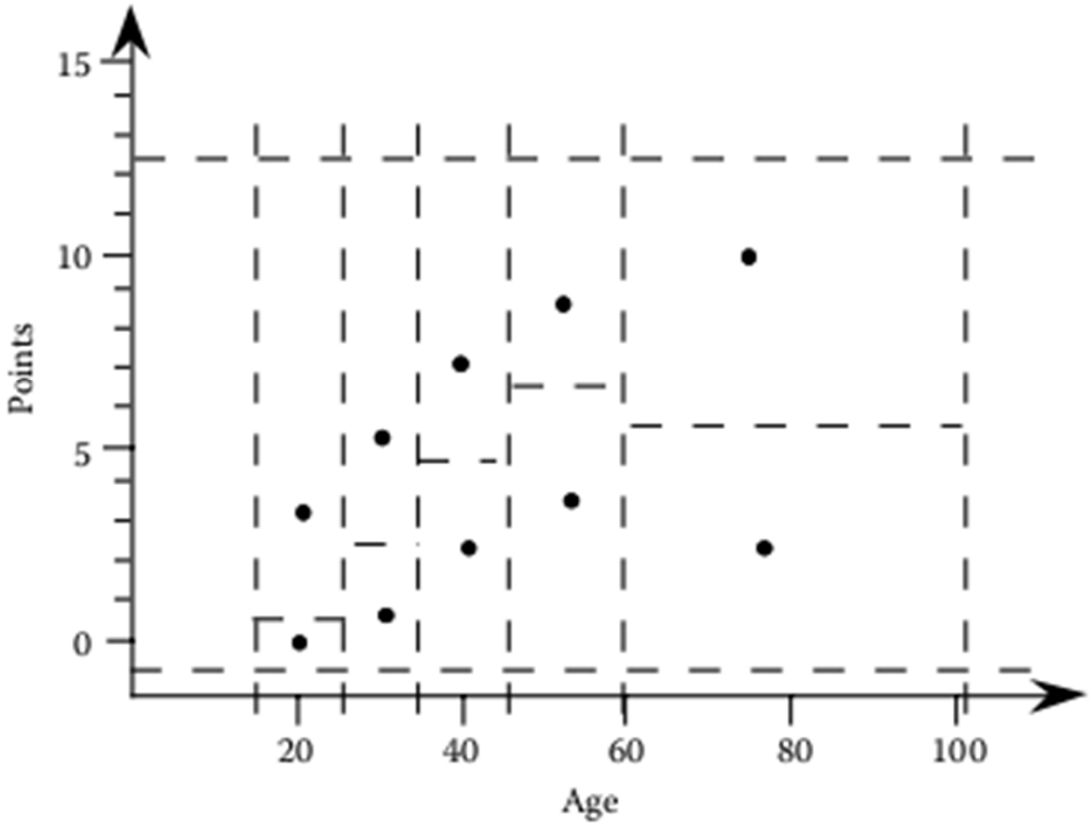
Jorgensen, Paul C. Software Testing
A Craftsman Approach

Decision Table Test Cases

Table 8.3 Decision Table Test Cases for the Insurance Premium Program

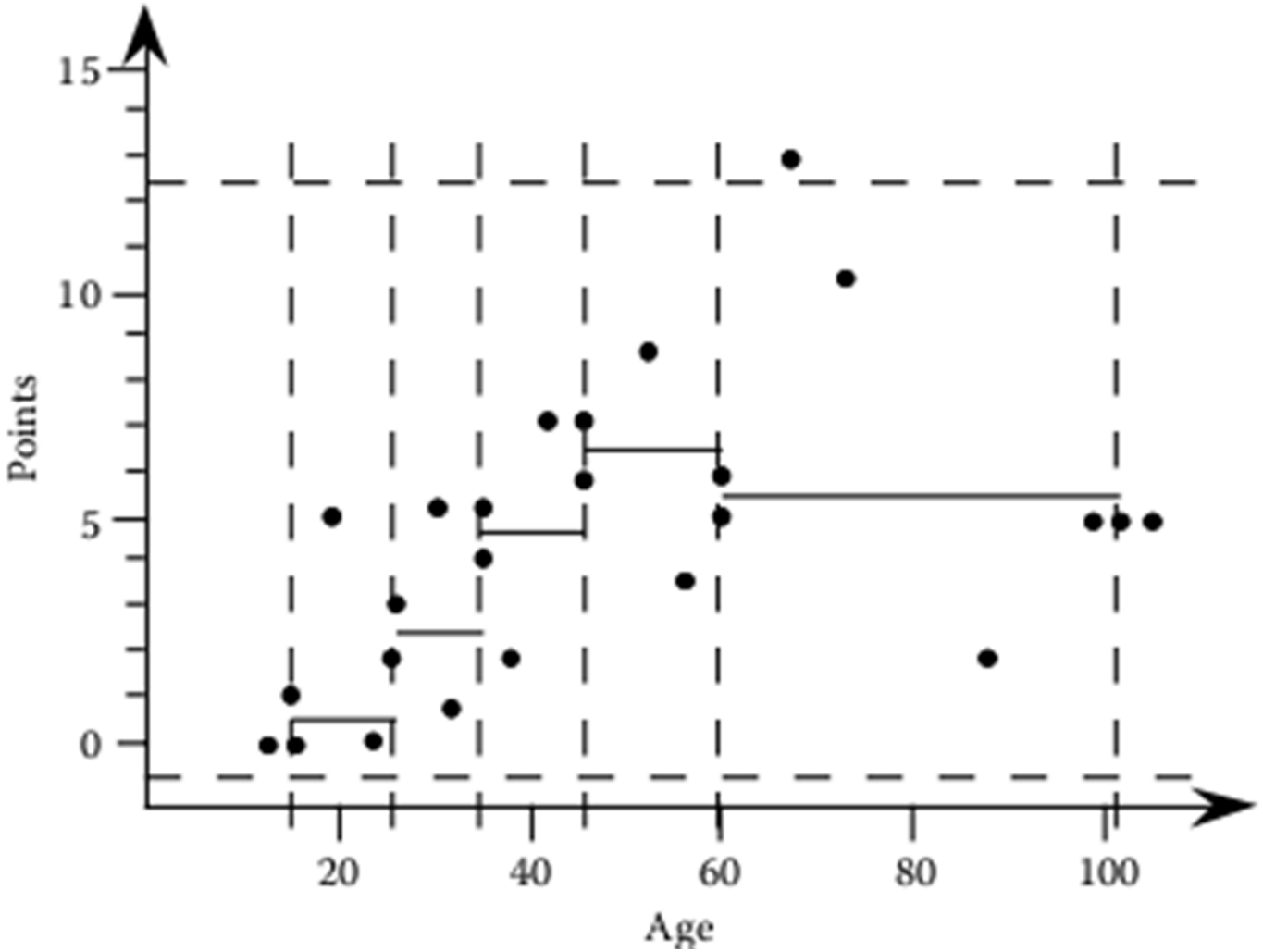
Age Is	*16–25*	*16–25*	*25–35*	*25–35*	*35–45*	*35–45*	*45–60*	*45–60*	*60-100*	*60-100*
Points	0	1–12	0–2	3–12	0–4	5–12	0–6	7–12	0–4	5–12
Age multiplier	2.8	2.8	1.8	1.8	1	1	0.8	0.8	1.5	1.5
Safe driving reduction	50	—	50	—	100	—	150	—	200	—

Decision Table Test Cases



Jorgensen, Paul C. Software Testing
A Craftsman Approach

Hybrid Test Cases



Jorgensen, Paul C. Software Testing
A Craftsman Approach