Presentation 9

Software testing - strategies

- Definitions and objectives
- Software testing strategies
- Software test classifications
- White box testing
  - Data processing and calculation correctness tests
  - Correctness tests and path coverage
  - Correctness tests and line coverage
  - McCabe's cyclomatic complexity metrics
  - Software qualification and reusability testing
  - Advantages and disadvantages of white box testing
- Black box testing
  - Equivalence classes for output correctness tests
  - Other operation factor testing classes
  - Revision factor testing classes
  - Transition factor testing classes
  - Advantages and disadvantages of black box testing

OHT 9.2

Software tests - definition

Software testing is a **formal** process carried out by a **specialized testing team** in which a software unit, several integrated software units or an entire software package are examined by **running the programs on a computer**. All the associated tests are performed according to **approved test procedures** on **approved test cases**.

OHT 9.3

Software testing objectives

Direct objectives

a. To identify and reveal as many errors as possible in the tested software
b. To bring the tested software, after correction of the identified errors and retesting, to an acceptable level of quality.
c. To perform the required tests efficiently and effectively, within the limits budgetary and scheduling limitation.

Indirect objectives

a. To compile a record of software errors for use in error prevention (by corrective and preventive actions)

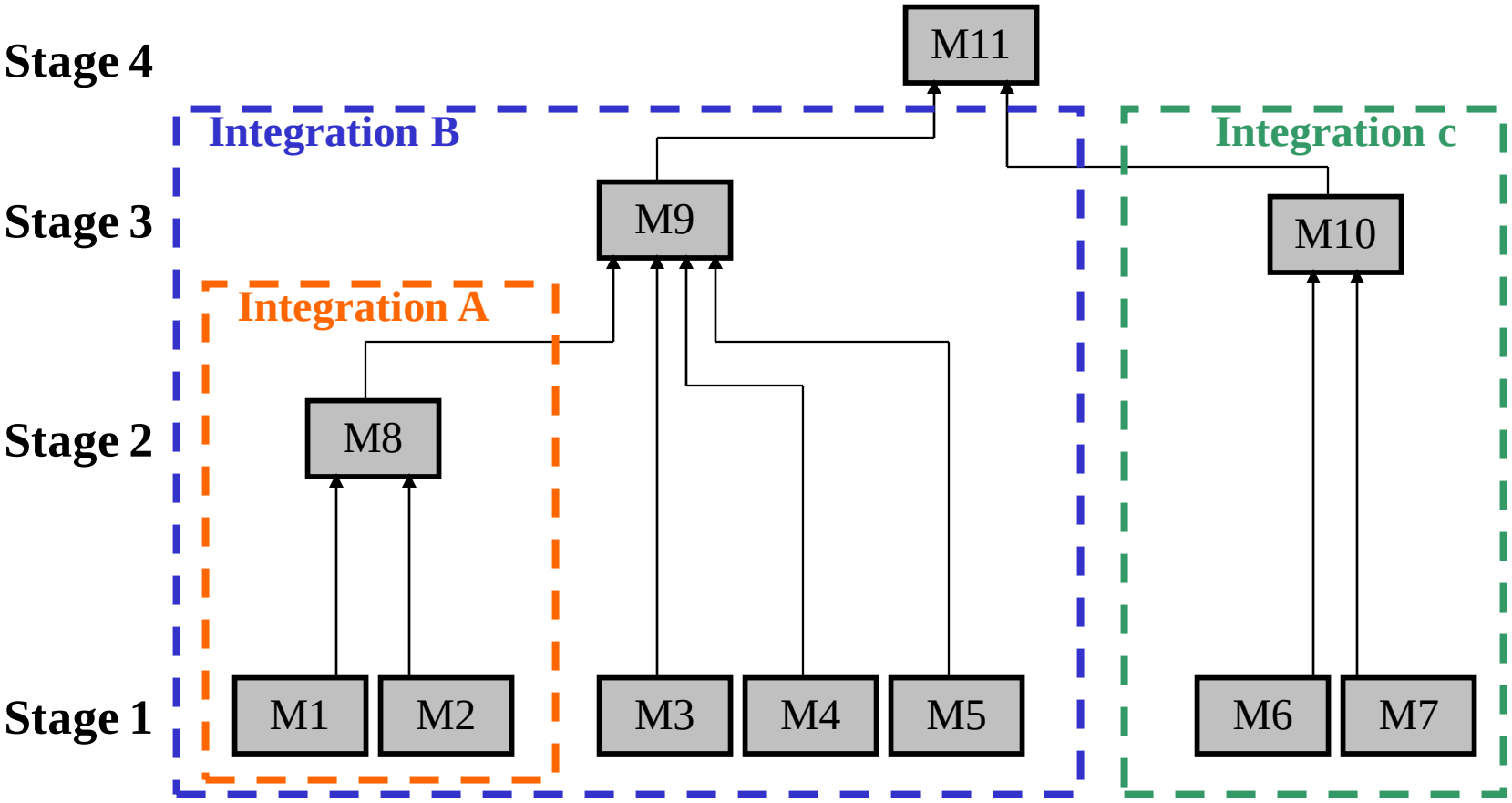
OHT 9.4

Software testing strategies

- ***Incremental testing strategies:***
  - Bottom-up testing
  - Top-down testing
- ***Big bang testing***

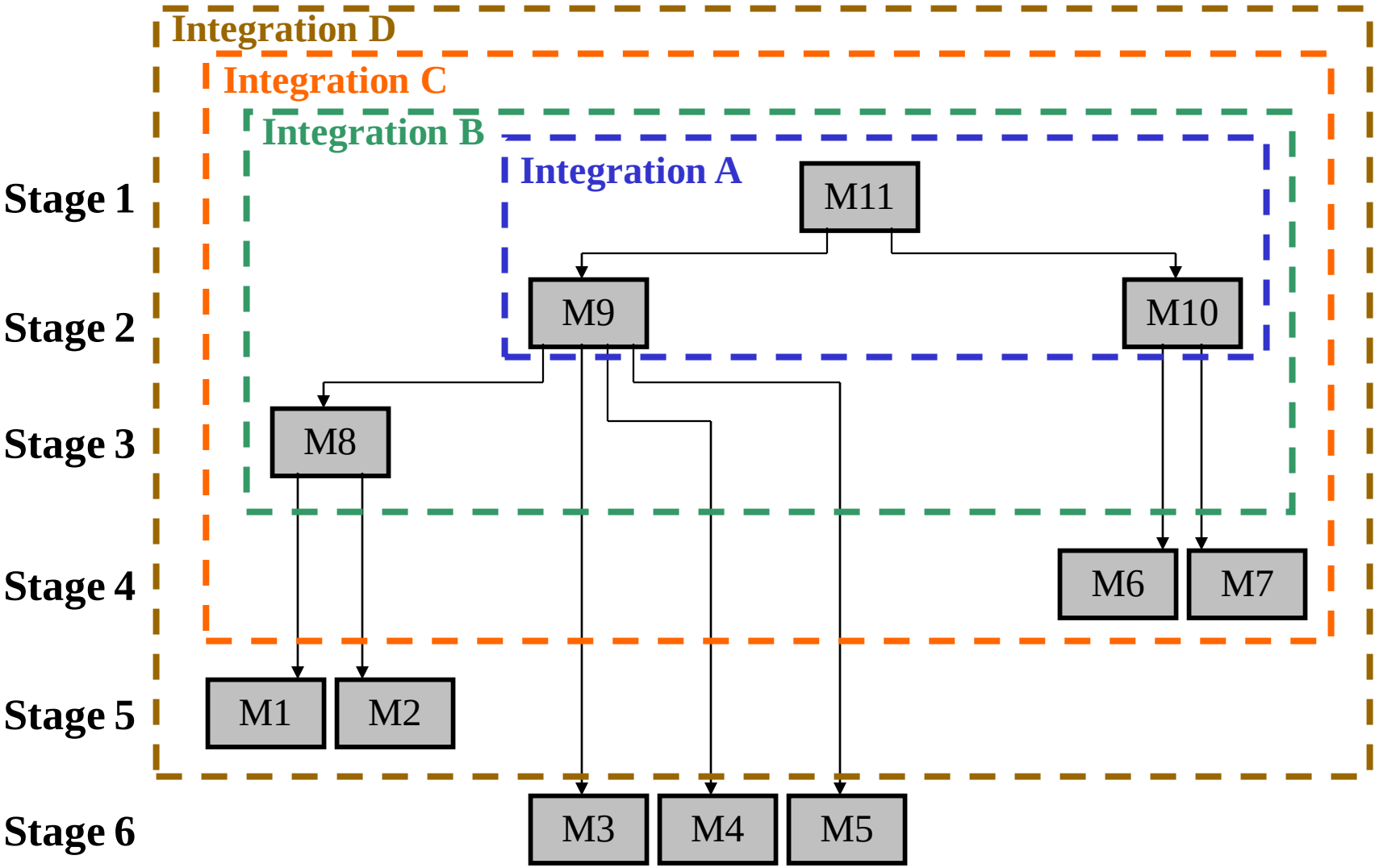
OHT 9.5

Bottom-up testing



OHT 9.6

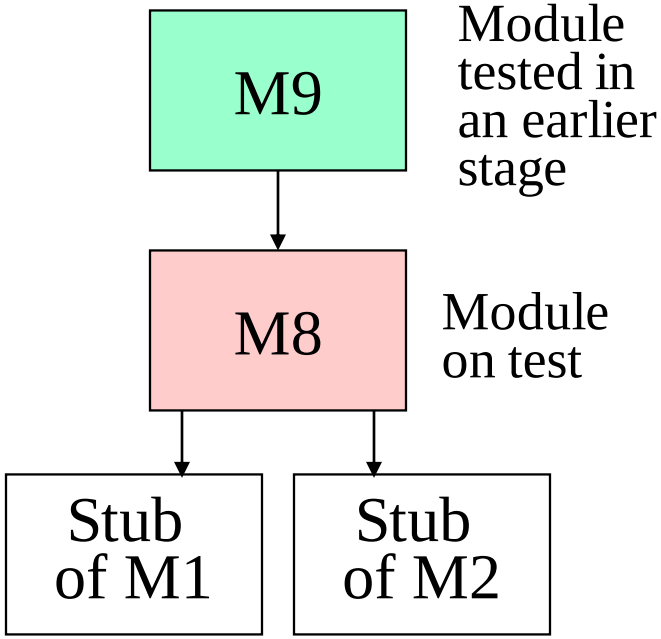
Top-down testing



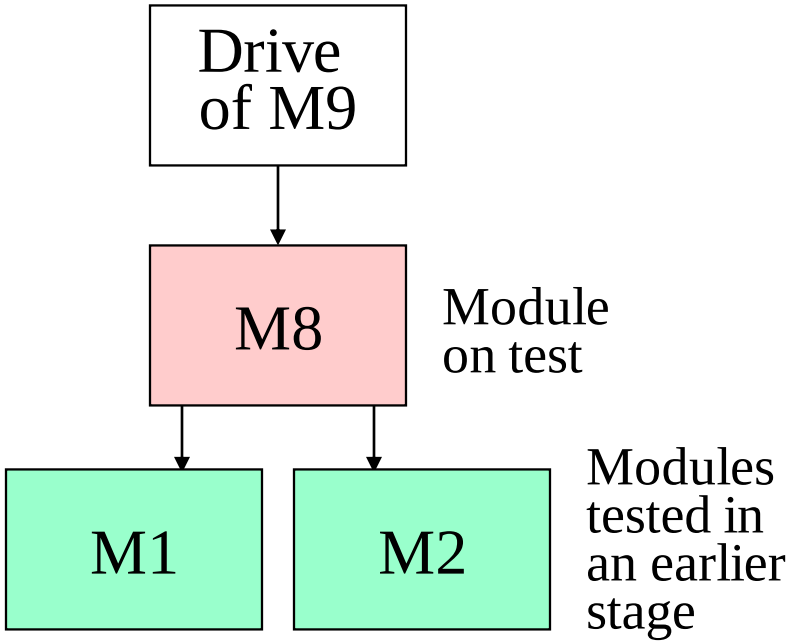
OHT 9.7

Use of stubs and drivers for incremental testing

Top-down testing of module M8



Bottom-up testing of module M8



OHT 9.8

Black box and white box - IEEE definitions

Black box testing

1. **Testing that ignores the internal mechanism of the system or component and focuses solely on the outputs in response to selected inputs and execution conditions**
2. **Testing conducted to evaluate the compliance of a system or component with specified functional requirements**

White box testing

Testing that takes into account the internal mechanism of a system or component

OHT 9.9

White box testing
"Path" vs "line" coverage

Path coverage

- Path coverage of a test is measured by the percentage of all possible program paths included in planned testing.

Line coverage

- Line coverage of a test is measured by the percentage of program code lines included in planned testing.

OHT 9.10

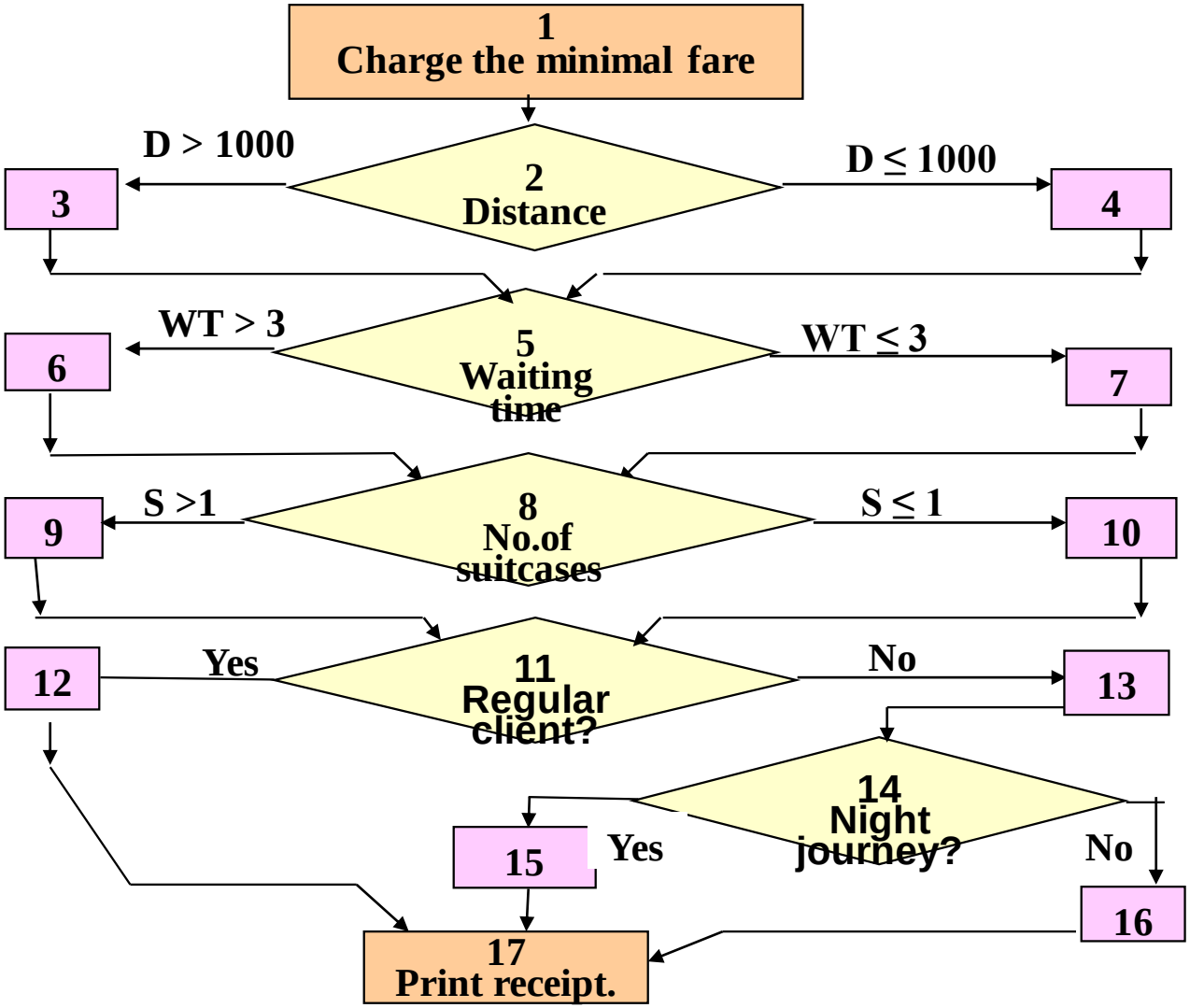
The Imperial Taxi Services (ITS) taximeter

EExample ITS taxi fares for one-time passengers are calculated as follows:

1. Minimal fare: $2. This fare covers the distance traveled up to 1000 yards and waiting time (stopping for traffic lights or traffic jams, etc.) of up to 3 minutes.
2. For every additional 250 yards or part of it: 25 cents.
3. For every additional 2 minutes of stopping or waiting or part thereof: 20 cents.
4. One suitcase: 0 change; each additional suitcase: $1.
5. Night supplement: 25%, effective for journeys between 21.00 and 06.00. Regular clients are entitled to a 10% discount and are not charged the night supplement.

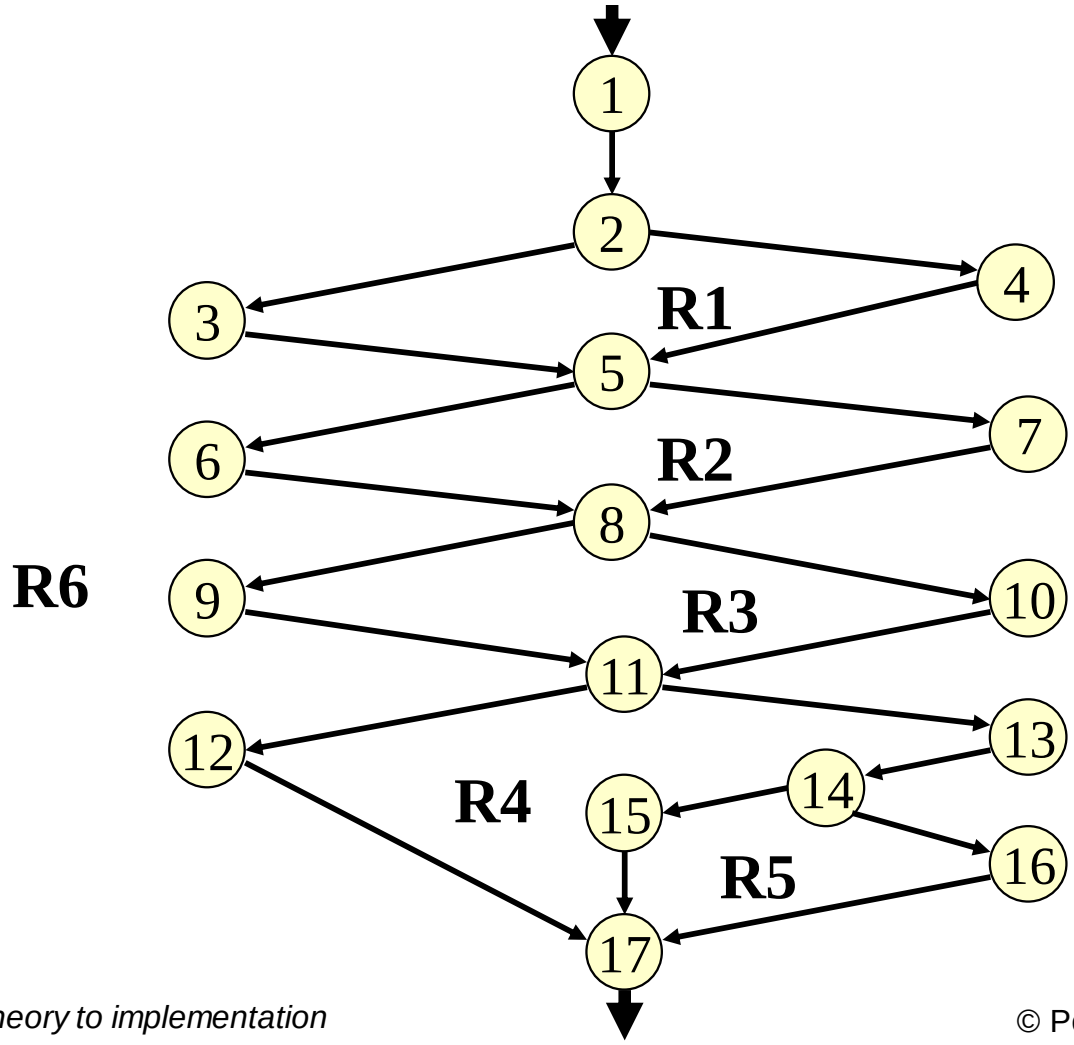
OHT 9.11

ITS - Flow chart



OHT 9.12

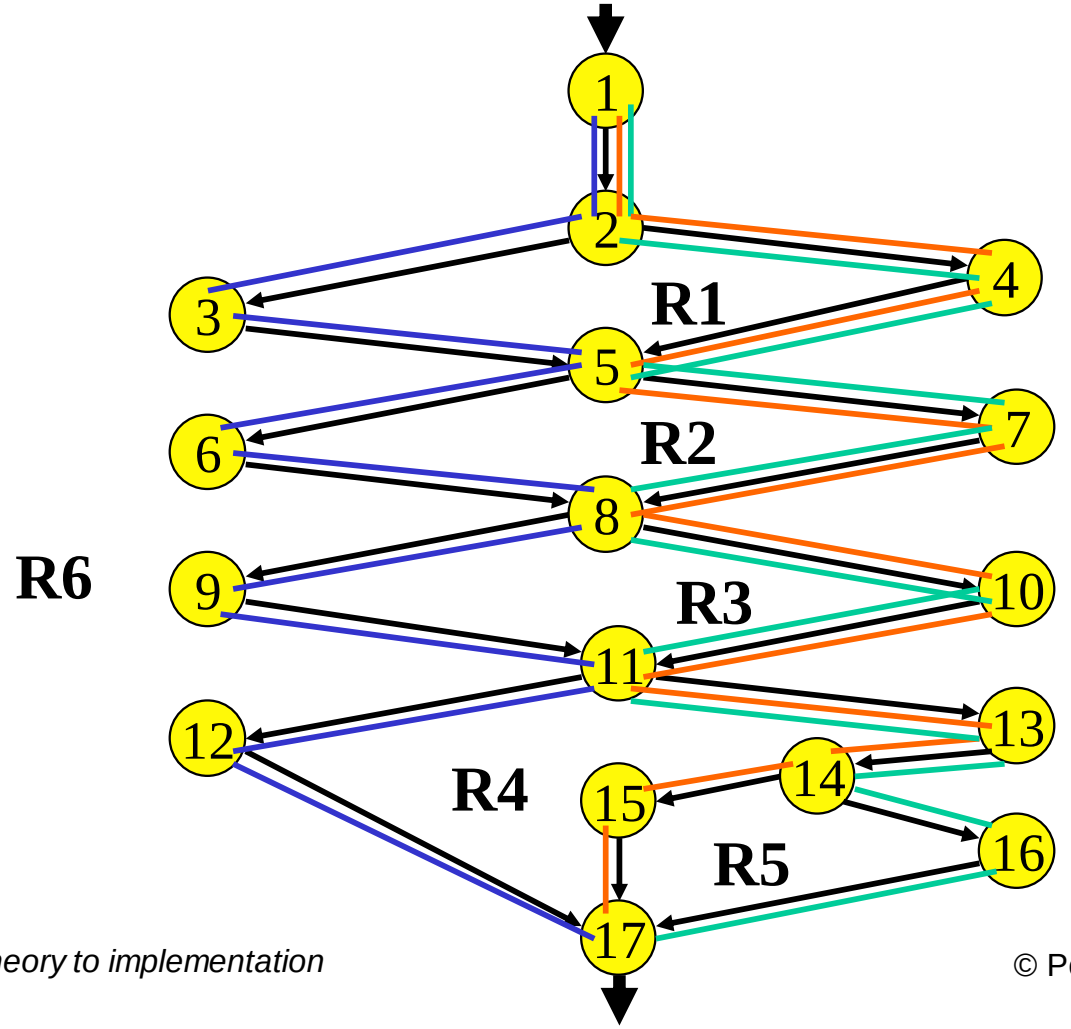
ITS - Program flow graph



OHT 9.13

ITS - The minimum number of paths

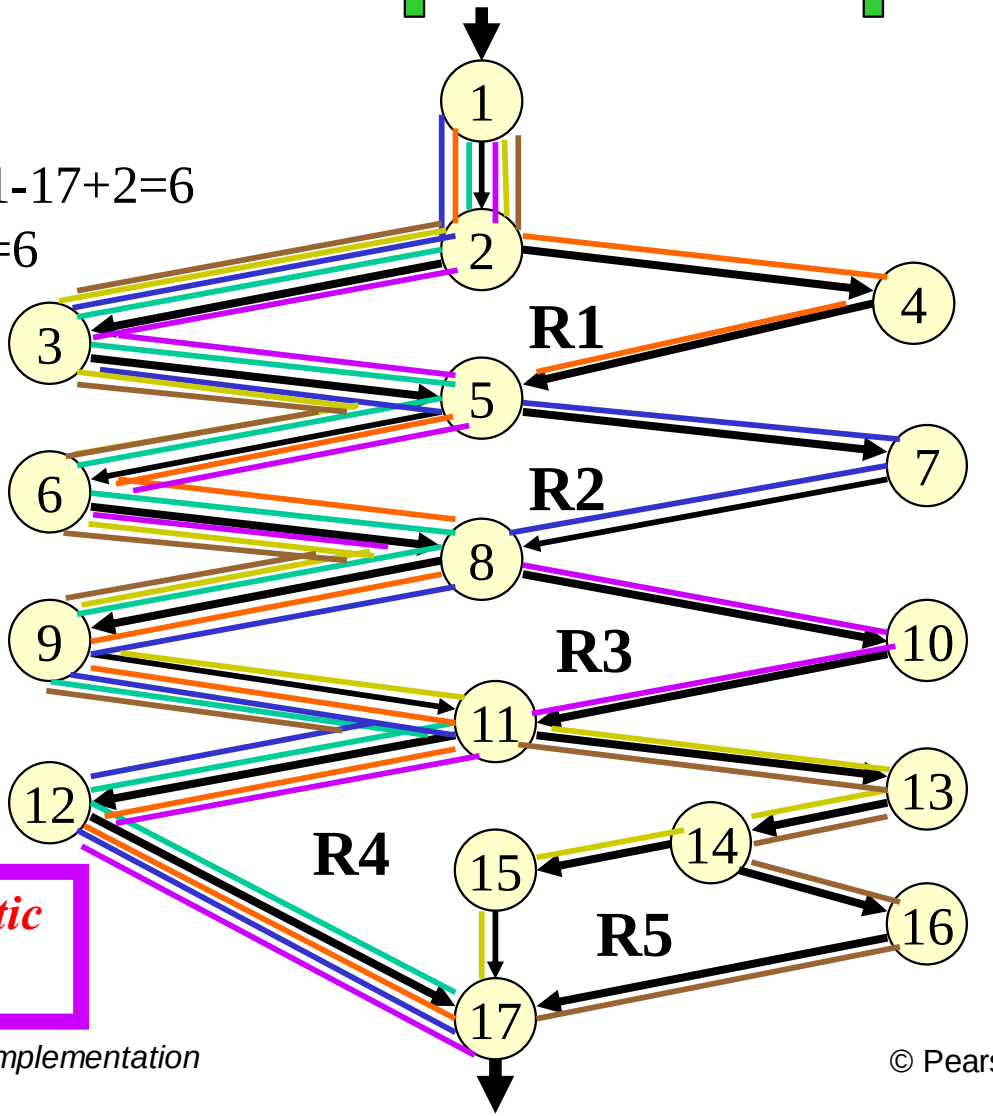
for full line coverage



OHT 9.14

ITS - The maximum set of independent paths

V(G)=R=6
V(G)=E-N+2=21-17+2=6
V(G)=P+1=5+1=6



R=Regions
N=Nodes
E=Edges
P=Decisions

McCabe's cyclomatic complexity metrics

OHT 9.15

Advantages and disadvantages of white box testing

Advantages:

* Direct determination of software correctness as expressed in the processing paths, including algorithms.
* Allows performance of line coverage follow up.
* Ascertains quality of coding work and its adherence to coding standards.

Disadvantages :

* The vast resources utilized, much above those required for black box testing of the same software package.
* The inability to test software performance in terms of availability (response time), reliability, load durability, etc.

OHT 9.16

Equivalence class partitioning (EC)

A black box method aimed at increasing the efficiency of testing and, at the same time, improving coverage of potential error conditions.

OHT 9.17

Equivalence class partitioning (EC)

- An equivalence class (EC) is a set of input variable values that produce the same output results or that are processed identically.
- EC boundaries are defined by a single numeric or alphabetic value, a group of numeric or alphabetic values, a range of values, and so on.
- An EC that contains only valid states is defined as a "valid EC," whereas an EC that contains only invalid states is defined as the "invalid EC."
- In cases where a program's input is provided by several variables, valid and invalid ECs should be defined for each variable.

OHT 9.18

Equivalence class partitioning (EC)

According to the equivalence class partitioning method:

- **E**ach valid EC and each invalid EC are included in at least one test case.
- **D**efinition of test cases is done separately for the valid and invalid ECs.
- **I**n defining a test case for the valid ECs, we try to cover as many as possible "new" ECs in that same test case.
- **I**n defining invalid ECs, we must assign one test case to each "new" invalid EC, as a test case that includes more than one invalid EC may not allow the tester to distinguish between the program's separate reactions to each of the invalid ECs.
- **T**est cases are added as long as there are uncovered ECs.

OHT 9.19

Entrance ticket price table - The Pool

Day	Mon., Tue., Wed., Thu, Fri.				Sat., Sun.			
Visitor's status	Ot	Ot	Mem	Mem	Ot	Ot	Mem	Mem
Entry hour	6.00-19.00	19.01-24.00	6.00-19.00	19.01-24.00	6.00-19.00	19.01-24.00	6.00-19.00	19.01 - 24.00
Age: 0.00-16.00	$5	$6	$2.50	$3	$7.50	$9	$3.50	$4
Age 16.01-60.00	$10	$17	$5	$6	$15	$18	$7	$8
Age: 60.01-120.00	$8	$8	$4	$4	$12	$12	$5.50	$5.50

OHT 9.20

Test cases - The ticket price module

Test case type	Test case no.	Day of week	Visitor's status	Entry hour	Visitor's age	Test case result
Valid ECs	1	Mon.	Ot	7.55	8.4	$5
	2	Sat.	Mem	20.44	42.7	$8
	3	Sat.	Mem	22.44	65.0	$5.50
	4	Sat.	Mem	6.00	0.0	$3.50
	5	Sat.	Mem	19.00	16.0	$3.50
	6	Sat.	Mem	19.01	16.01	$8
	7	Sat.	Mem	19.01	60.0	$8
	8	Sat.	Mem	24.00	60.01	$5.50
	9	Sat.	Mem	24.00	120.0	$5.50
Invalid ECs	10	Mox.	Ot	7.55	8.4	Invalid day
	11	Mon.	88	7.55	8.4	Invalid visitor's status
	12	Mon.	Ot	4.40	8.4	Invalid entry hour
	13	Mon.	Ot	8@	8.4	Invalid entry hour
	14	Mon.	Ot	7.55	TTR	Invalid visitor's age
	15	Mon.	Ot	7.55	150.1	Invalid visitor's age

OHT 9.21

Advantages and disadvantages of black box testing

Advantages:

* **Allows us to carry out the majority of testing classes, most of which can be implemented solely by black box tests, i.e. load tests and availability tests.**
* **For testing classes that can be carried out by both white and black box tests, black box testing requires fewer resources.**

Disadvantages:

* **Possibility that coincidental aggregation of several errors will produce the correct response for a test case, and prevent error detection.**
* **Absence of control of line coverage. There is no easy way to specify the parameters of the test cases required to improve coverage.**
* **Impossibility of testing the quality of coding and its strict adherence to the coding standards.**