



SOFTWARE TESTING

苏临之

sulinzhi029@nwu.edu.cn



Examination

1. True or false questions.
2. Term explanation
3. Short answer questions.
4. Integrative questions.



True or False Questions

1. Software fault may not lead to a failure. ✓
2. Software testing will occupy about one third of the developing process. ✗
3. In the bottom-up testing, the stub modules will be established. ✗
4. In the static black-box testing, the specification is mainly examined. ✓



True or False Questions

- 5. The tester should never be too careful in the testing process. ✓
- 6. The tools for testing cannot take the place of artificial testing. ✓
- 7. A good tester relentlessly strives for perfection. ✗
- 8. The incremental testing does better than the non-incremental testing in concurrency. ✗



True or False Questions

- 9. If the product development is in a hurry, one can skip unit testing and proceed directly to integration testing. ✖
- 10. The more bugs one has found, the less bugs he will find. ✖



Term Explanation

1. Software defect

❖ 满足以下5个规则之一可判定为软件缺陷：

- 软件未实现产品说明书要求的功能；
- 软件出现了产品说明书指明不应出现的错误；
- 软件实现了产品说明书未提到的功能；
- 软件未实现产品说明书虽未明确提及但应该实现的目标；
- 软件难以理解、不易使用、运行缓慢或者从测试员的角度看最终用户会认为不好；



Term Explanation

2. Pesticide paradox

- 杀虫剂悖论指的是对软件进行越多的测试，那么该软件对软件测试人员的测试就越具有免疫力。

3. Equivalence class

- 将测试空间划分成若干个子集，如果每个子集中的任一数据对揭露程序中的缺陷都被认为是等价的，这些子集就叫做等价类。



Term Explanation

4. Test-to-pass and test-to-fail

- 通过性测试指的是利用最简单最直观的测试用例，确认软件基本功能。
- 失效性测试指的是纯粹为破坏软件而设计和执行的测试，用来蓄意攻击软件的薄弱环节。



Term Explanation

5. Test case

- 测试用例是为某个特殊目的而编制的一组输入数据，用于测试输出、执行条件以及预期后果，以便测试某个顺序途径或核实能否满足某个特定需求。



Short Answer Questions

1. When should we turn to the decision table testing?

- 当被测软件输入变量之间存在逻辑关系或输入与输出之间存在因果关系时，就可以使用判定表。

Short Answer Questions

2. What are the benefits of static white-box testing?

- 静态白盒测试在开发过程早期发现软件缺陷，使修复的时间和费用大幅降低。软件测试员可以得到软件如何运做的信息，存在哪些弱点和危险，而且可以与程序员建立良好的伙伴关系。项目状态可以传达给参与测试的所有小组成员。



Short Answer Questions

3. All the found defects do not necessarily need fixing.
Please show the reasons.

- 没有足够的时间修复缺陷
- 并不是真正的缺陷
- 修复缺陷风险太大
- 不值得修复



Short Answer Questions

4. Why would testers usually prefer spiral model?

- 测试员可以很早就参与开发过程，并且有机会尽早发现问题，为项目节省更多成本。



Short Answer Questions

5. Why must we fix the defects as early as possible?

- 因为越早修复缺陷，修复缺陷的代价就越小。
- 等价答案：因为越晚修复缺陷，修复缺陷的代价就越大。



Testing Techniques

1. A website of a school will show the graduate students enrolled from 1995 to 2020 through a query of the system. The post-graduate students will be enrolled only in autumn (00), and the doctoral students will be enrolled in both autumn (01) and spring (11). In the system, a 6-digit character string, whose 4 higher digits represent the year ranging from 1995 to 2020 and 2 lower digits must be 00, 01 or 11, will be input into the dialog. Please use the equivalence partitioning method to show the valid and invalid equivalence classes, and then show the corresponding test cases.



Testing Techniques

1. A website of a school will show the graduate students enrolled from 1995 to 2020 through a query of the system. The post-graduate students will be enrolled only in autumn (00), and the doctoral students will be enrolled in both autumn (01) and spring (11). In the system, a 6-digit character string, whose 4 higher digits represent the year ranging from 1995 to 2020 and 2 lower digits must be 00, 01 or 11, will be input into the dialog. Please use the equivalence partitioning method to show the valid and invalid equivalence classes, and then show the corresponding test cases.



Testing Techniques

2. Here is a function $F = y - x$ with two integer variables x which varies from 10 to 30 and y which varies from 20 to 30. Please draw the graph used for the boundary value analysis and then give out the corresponding test cases by filling in the table below.

Test case number	x	y	F
.....



Testing Techniques

3. A repair station provides repair service for the mechanical machines. A machine will enjoy priority service if the following two conditions are both met:

(I) the power over 50 horsepower;

(II) with incomplete maintenance record, or with its operation over 10 years.

The other machines will be dealt with according to the application sequence. Please show the decision table for the issue and simplify it if necessary.



Testing Techniques

3. A repair station provides repair service for the mechanical machines. A machine will enjoy priority service if the following two conditions are both met:

(I) the power over 50 horsepower;

(II) with incomplete maintenance record, or with its operation over 10 years.

The other machines will be dealt with according to the application sequence. Please show the decision table for the issue and simplify it if necessary.



Testing Techniques

条件桩:

C1: the power over 50 horsepowers

C2: incomplete maintenance record

C3: operation over 10 years

动作桩:

A1: priority service

A2: according to the application sequence

Testing Techniques

原始判定表：

		1	2	3	4	5	6	7	8
条件部分	C1	T	T	T	T	F	F	F	F
	C2	T	T	F	F	T	T	F	F
	C3	T	F	T	F	T	F	T	F
动作部分	A1	√	√	√					
	A2				√	√	√	√	√

Testing Techniques

简化判定表：

		1	2	3	4	5
条件部分	C1	T	T	T	T	F
	C2	T	T	F	F	-
	C3	T	F	T	F	-
动作部分	A1	√	√	√		
	A2				√	√



Testing Techniques

4. The national post-graduate entrance examination will be held in a university. The teaching staffs and academic staffs will be required to be examination invigilators. The teaching staffs who have worked for over 3 years will go to the north campus, and who have worked for less than 3 years will go to the south campus. The academic staffs will all go to the south campus and those who are under the age of 35 will also assist the examination management. In addition, the security staffs are required to maintain public order throughout the examination. Please draw the cause-effect diagram for the issue.



Testing Techniques

4. The national post-graduate entrance examination will be held in a university. The teaching staffs and academic staffs will be required to be examination invigilators. The teaching staffs who have worked for over 3 years will go to the north campus, and who have worked for less than 3 years will go to the south campus. The academic staffs will all go to the south campus and those who are under the age of 35 will also assist the examination management. In addition, the security staffs are required to maintain public order throughout the examination. Please draw the cause-effect diagram for the issue.

Testing Techniques

c_1 : the teaching staffs

c_2 : the academic staffs

c_3 : over 3 years

c_4 : under 35

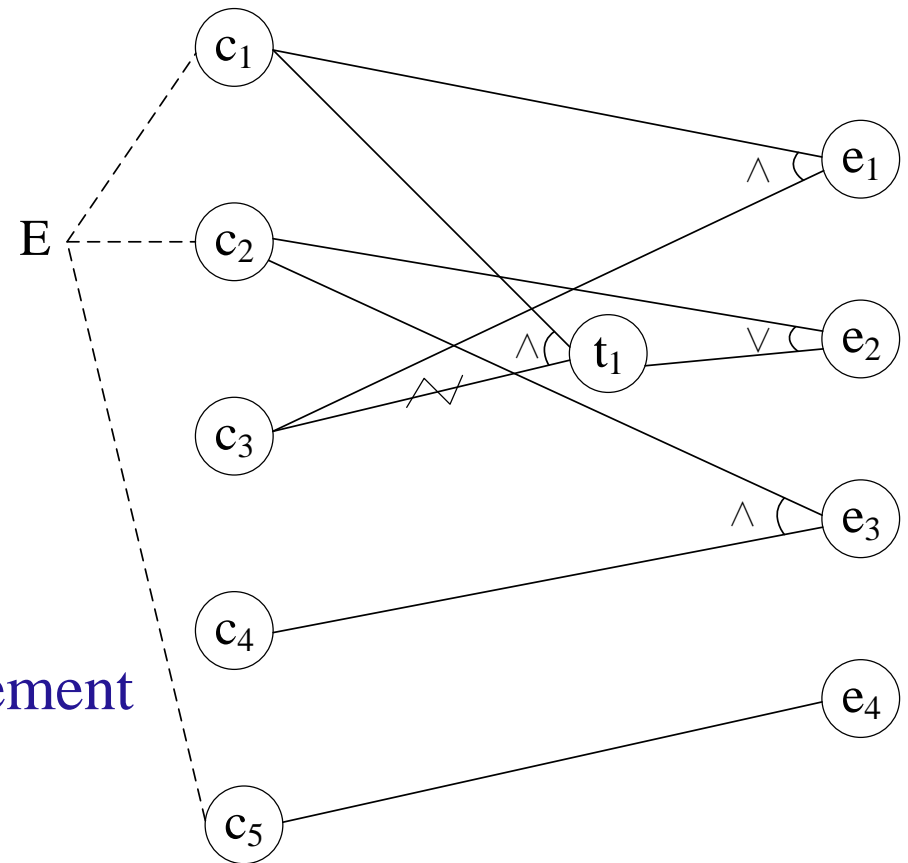
c_5 : the security staffs

e_1 : go to the north campus

e_2 : go to the south campus

e_3 : assist the examination management

e_4 : maintain public order



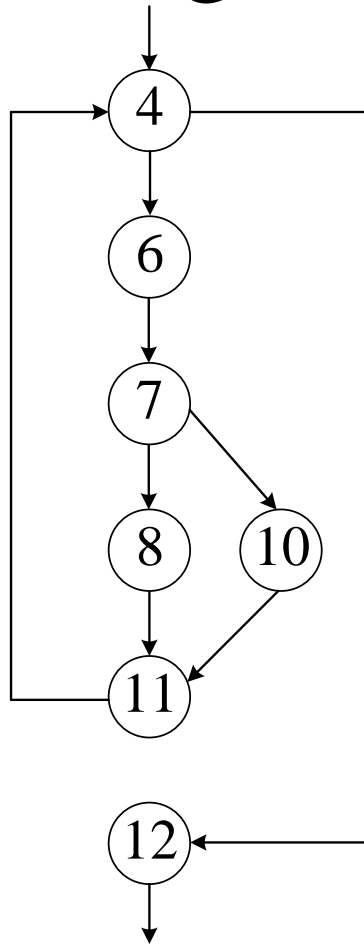
Testing Techniques

```
1  #include<stdio.h>
2  int main()
3  {
4      int a[10] = {0,1,2,3,4,5,6,7,8,9};
5      int b[10];
6      for (int i = 0; i <= 10; i++){
7          if a[i] <= 5;
8              b[i] = 0;
9          else
10             b[i] = 1;
11     }
12 }
```

Please draw the flow graph for the issue and calculate its cyclomatic complexity number (V). Then show the basic paths that comprise a basic path set.

Note: Label each node with the line number. If several lines correspond to one node, label the node with the first line number.

Testing Techniques



$$V = 3$$

基本路径集:

4-12

4-6-7-8-11-4-12

4-6-7-10-11-4-12



Testing Techniques

6. Please read the following code segment written in C programming language.

```
scanf ("%d %d", &s, &t) ;  
if (s > 2 || t > 3)  
    t = t + s;  
if (s == 4 && t == 8)  
    t = t + 2;
```

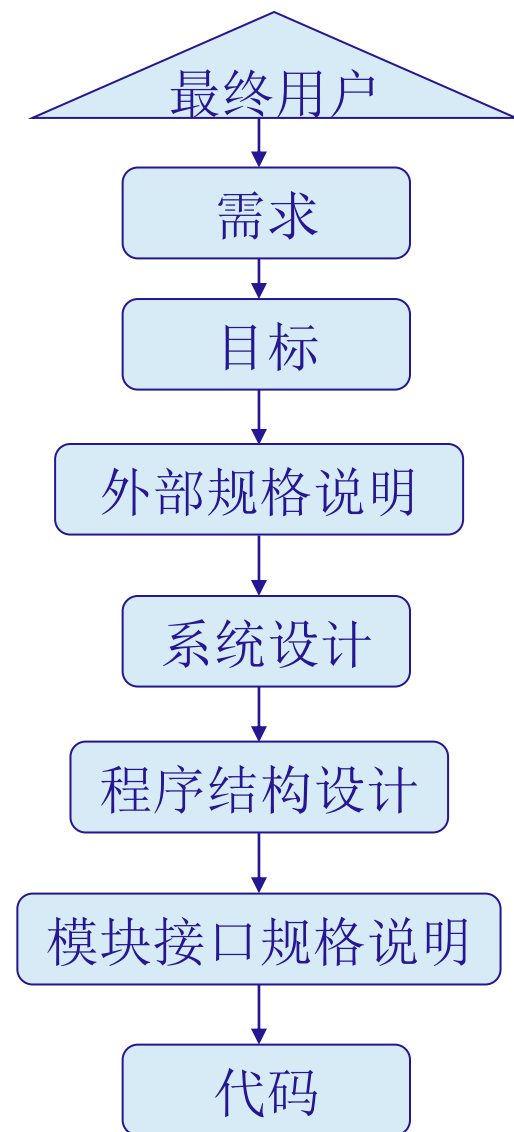
Please design test cases for the input variables by using the multiple condition coverage method and point out the corresponding combinations of conditions.

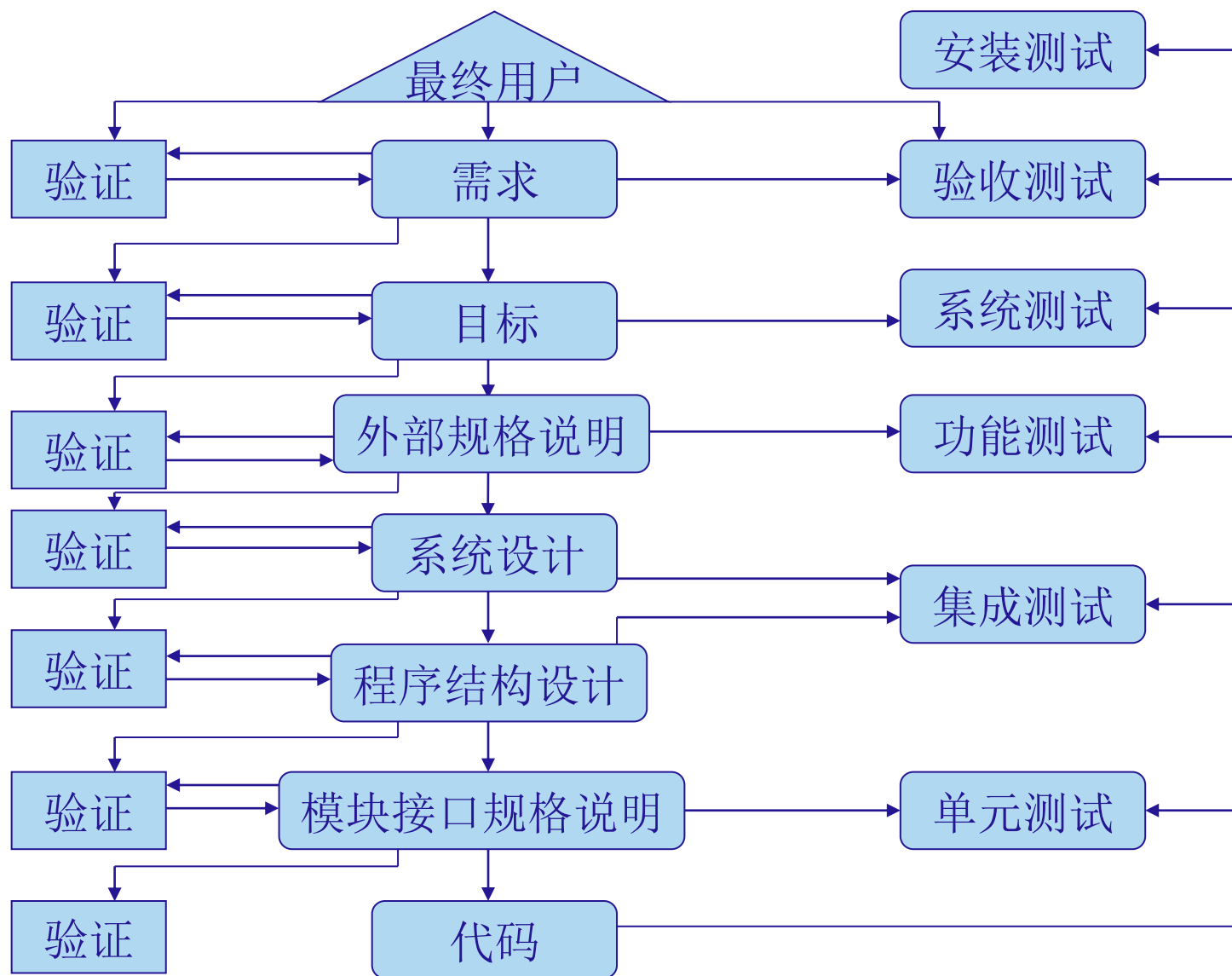


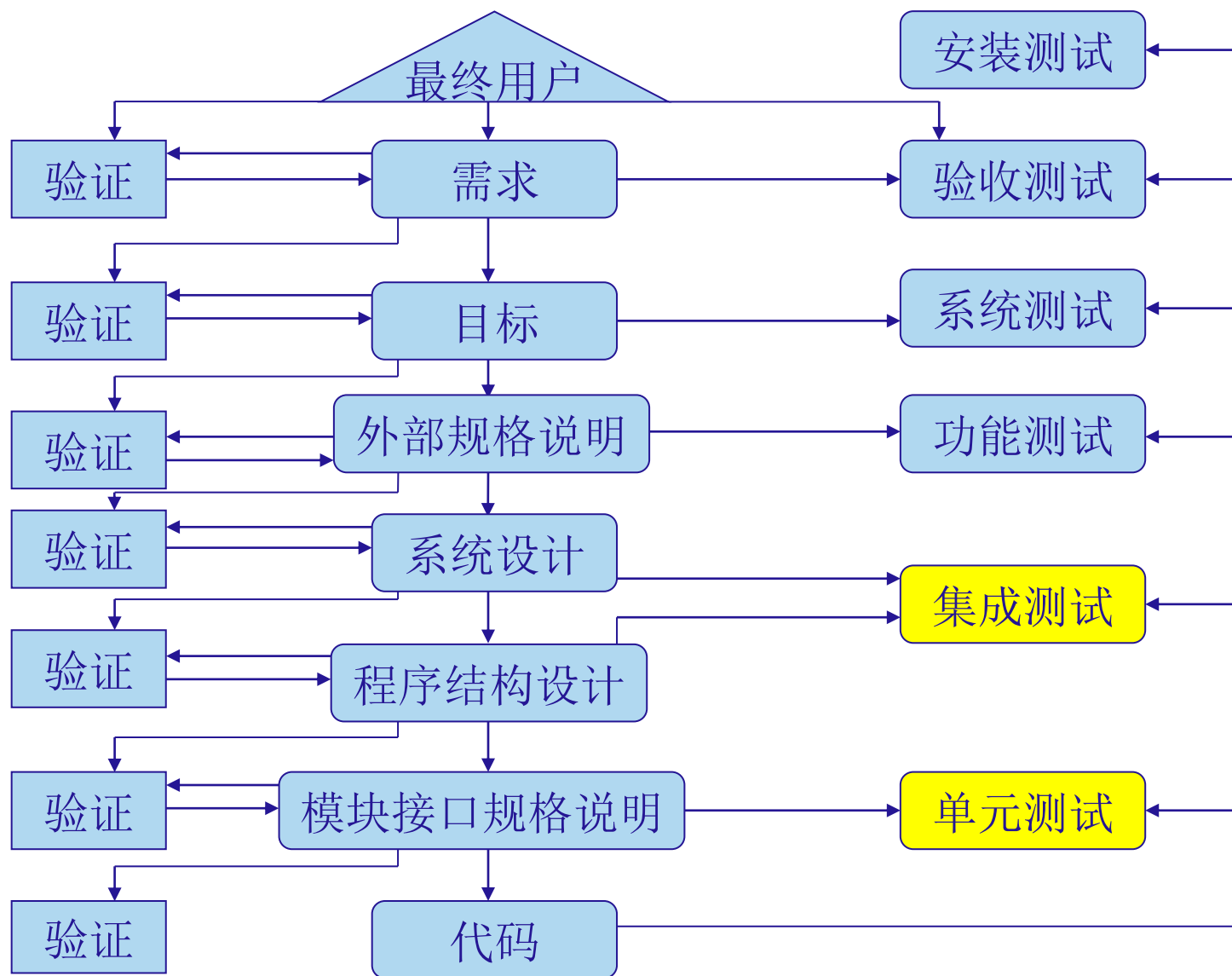
Testing Techniques

测试用例	覆盖的条件组合
$s = 4, t = 4$	T_1T_2, T_3T_4
$s = 2, t = 3$	F_1F_2, F_3F_4
$s = 4, t = 3$	T_1F_2, T_3F_4
$s = 2, t = 6$	F_1T_2, F_3T_4

Development of Software





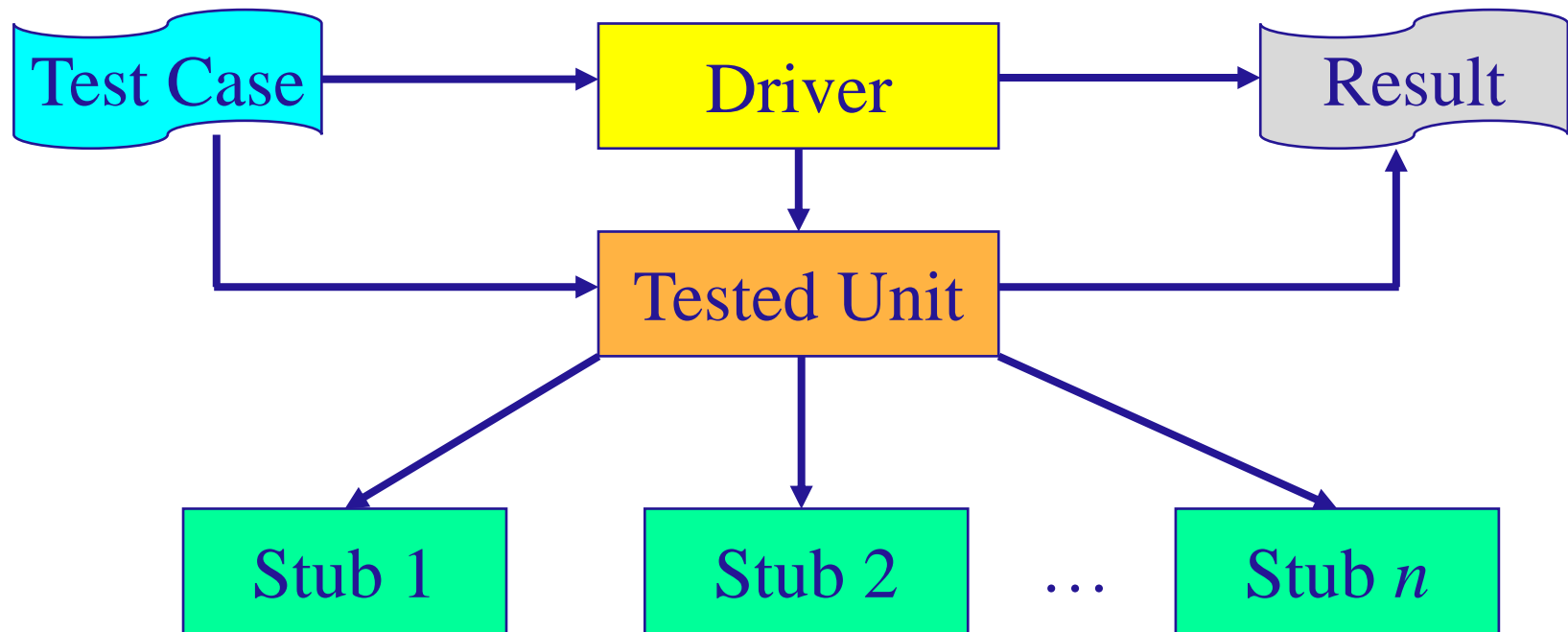




Unit Testing

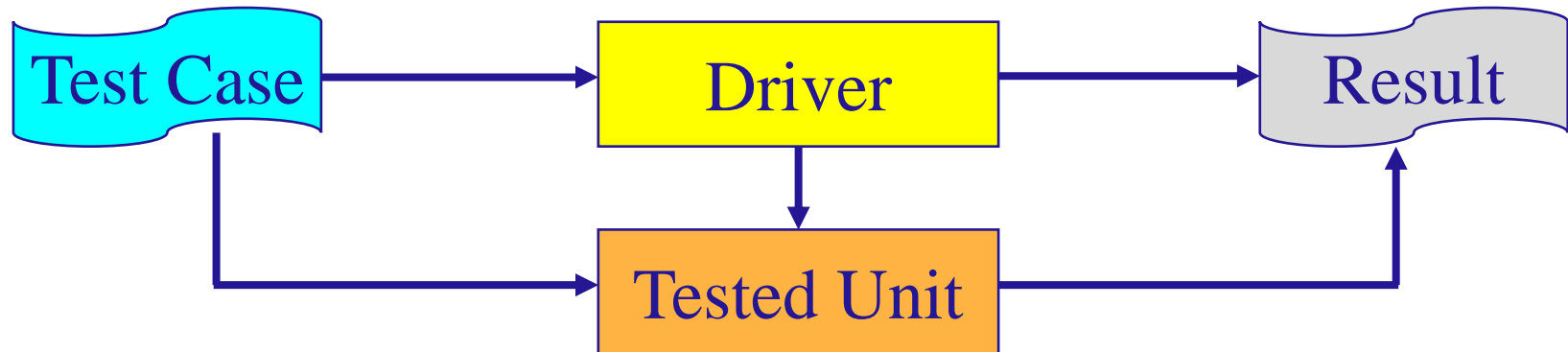
- Unit testing is also called the module testing, which alleviates the difficulty in the testing process. One can also test several units in parallel.
- The unit testing is generally a white-box testing, which means that it is essential to know the basic code in the module, including the input and output as well as its function.

Basic Model



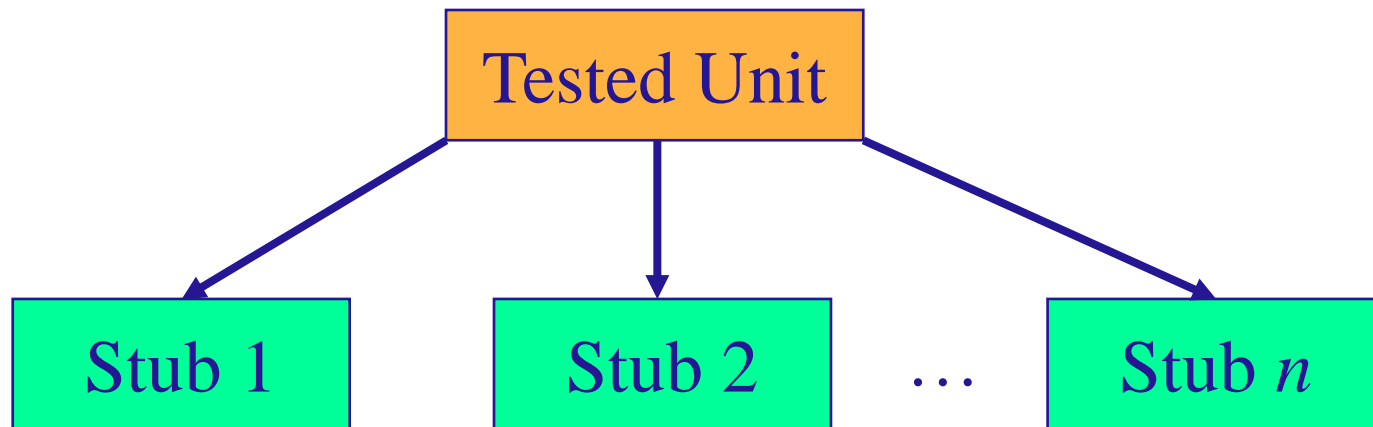
Driver Module

- The driver module acts as the higher part of the unit. It sends data to the unit and then shows the corresponding result after operation.



Stub Module

- The stub module acts as the lower sub-routine of the unit. By simulating the module called by the tested unit, the stub module itself does not operate but just return a static value.





Testing Methods

- Non-Incremental Testing: Also called the Big-bang integration. The units are tested individually and then integrated together for the integration testing.
- Incremental Testing: The tested modules are arranged into the structure of the program in advance. In this way the module testing also includes the integration testing, which will not serve as an individual part.
 - Top-Down Testing
 - Bottom-Up Testing



Comparison

Incremental Testing	Non-Incremental Testing
工作量小：使用前面测试过的模块来取代非增量测试中所需要的驱动模块或桩模块	工作量较大：要不断设计驱动模块和桩模块
可以较早发现模块中与不匹配接口、不正确假设等编程错误	到了测试过程的最后阶段，模块之间才能“互相看到”
容易进行调试，新出现的错误往往与最近添加的模块有关	直到整个程序组装之后，模块之间接口相关的错误才会浮现，难以定位

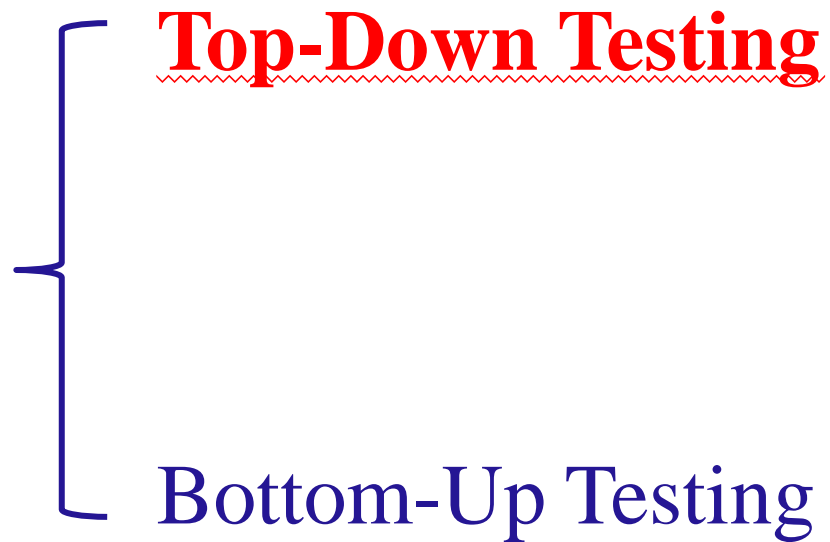


Comparison

Incremental Testing	Non-Incremental Testing
测试可以进行地更彻底，每个模块经受了更多的检验	使用驱动模块和桩模块而非实际模块，对被测试模块的测试只影响自身
在测试上花费的时间多，设计驱动模块和桩模块所用时间少	测试时间少，但设计驱动模块和桩模块需要大量时间
并行性差	可以同时并行测试很多模块



Incremental Testing

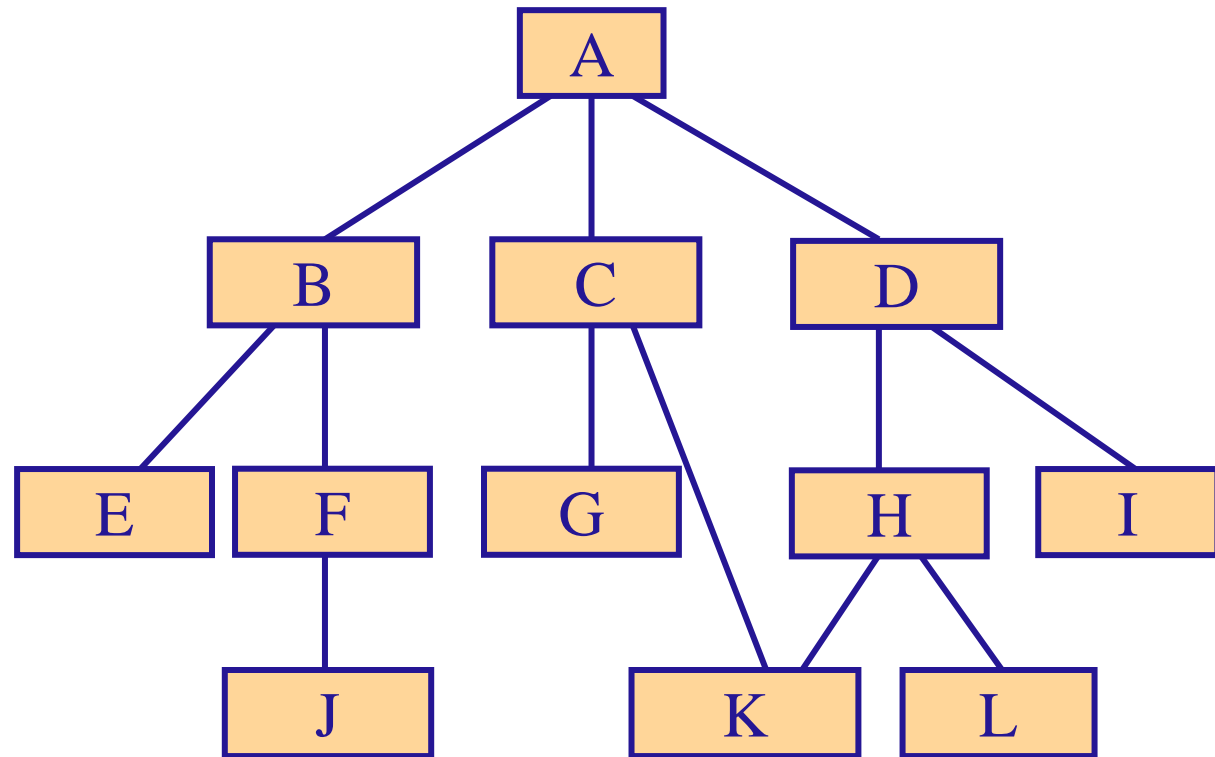




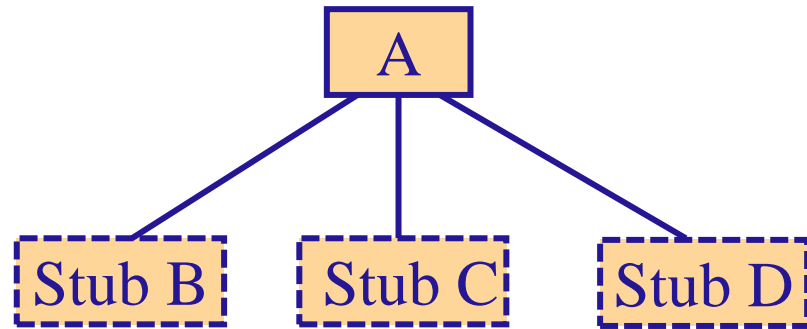
Top-Down Testing

- The top-down testing starts from the root node of the entire program tree. In the process, at least one module that calls the tested unit has been tested.
- To test the upper unit, the stub modules are established.

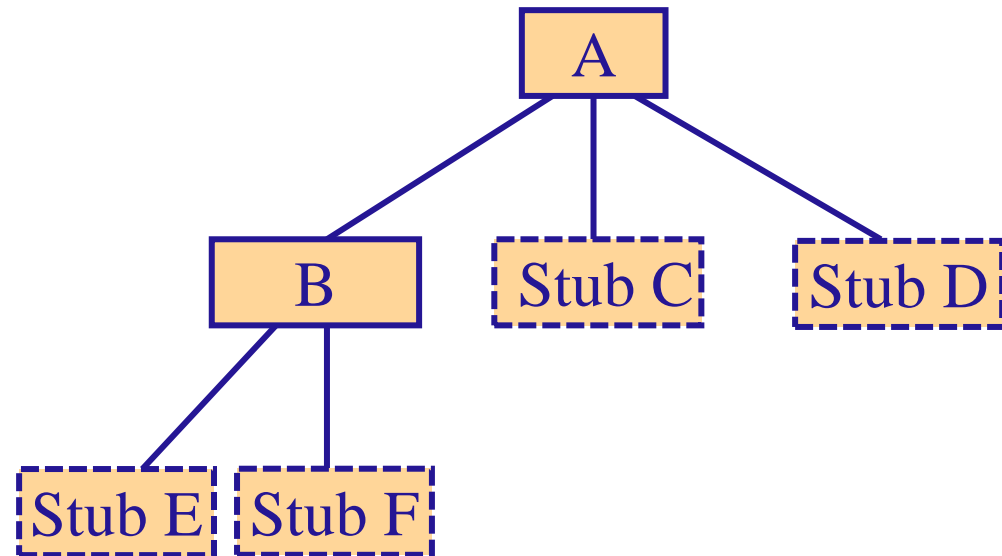
Top-Down Testing



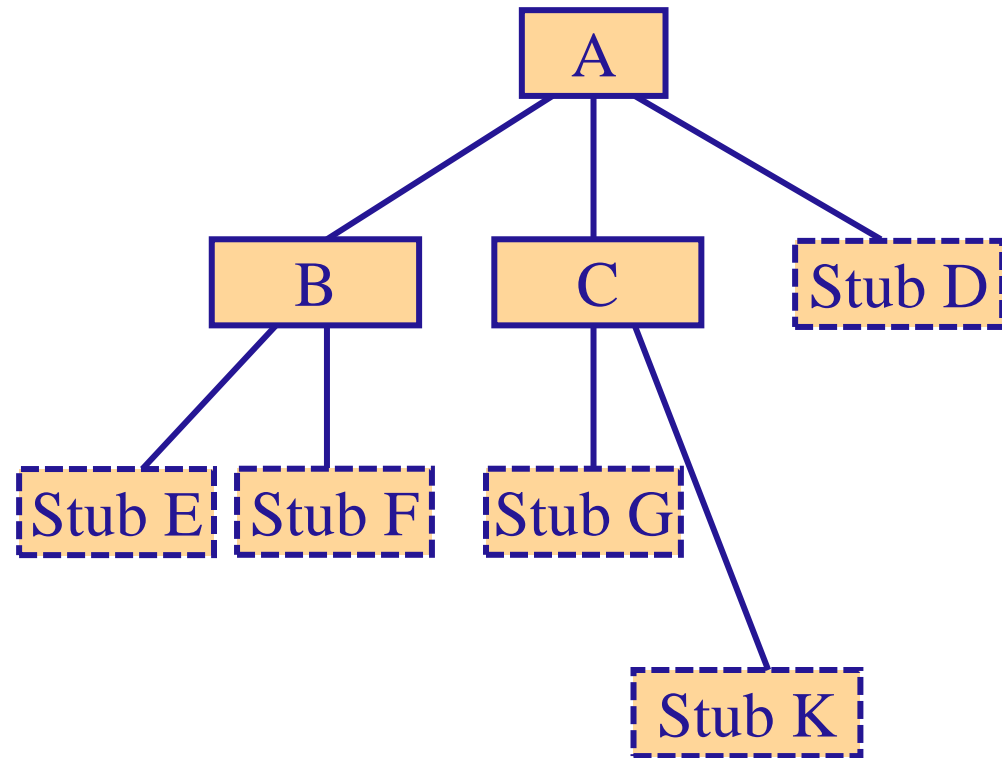
Top-Down Testing



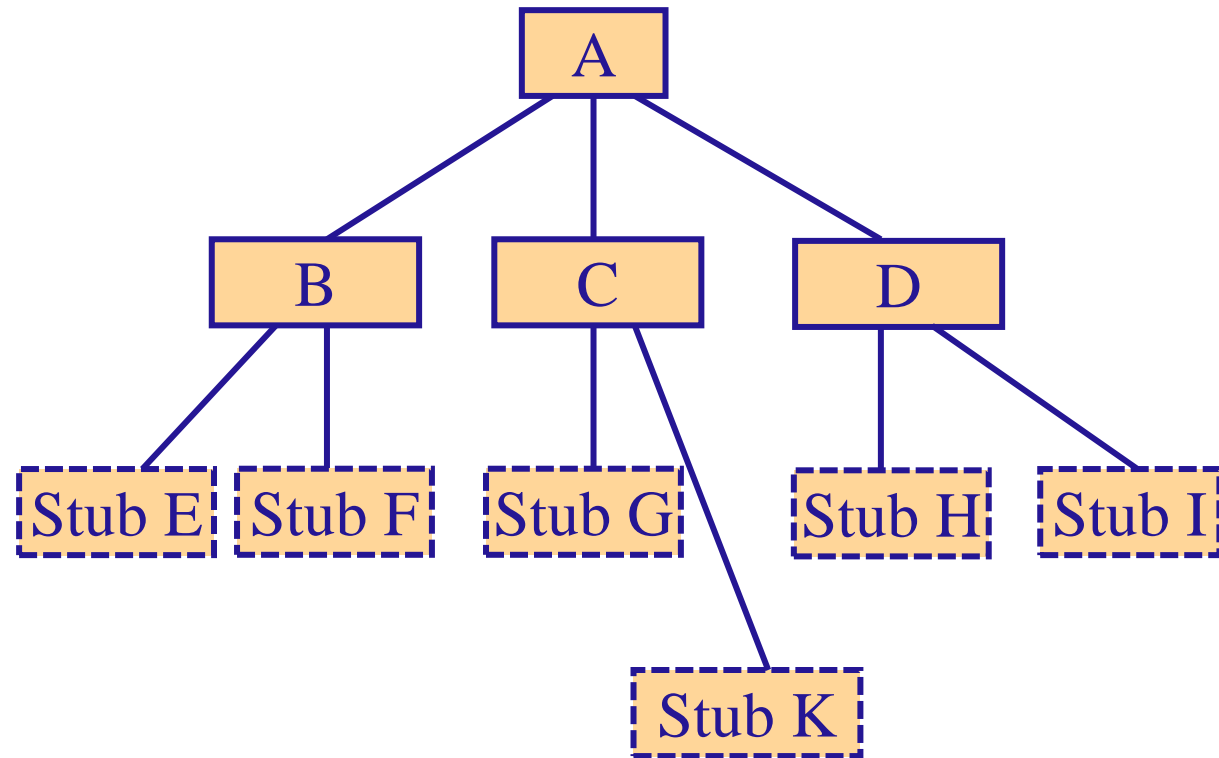
Top-Down Testing



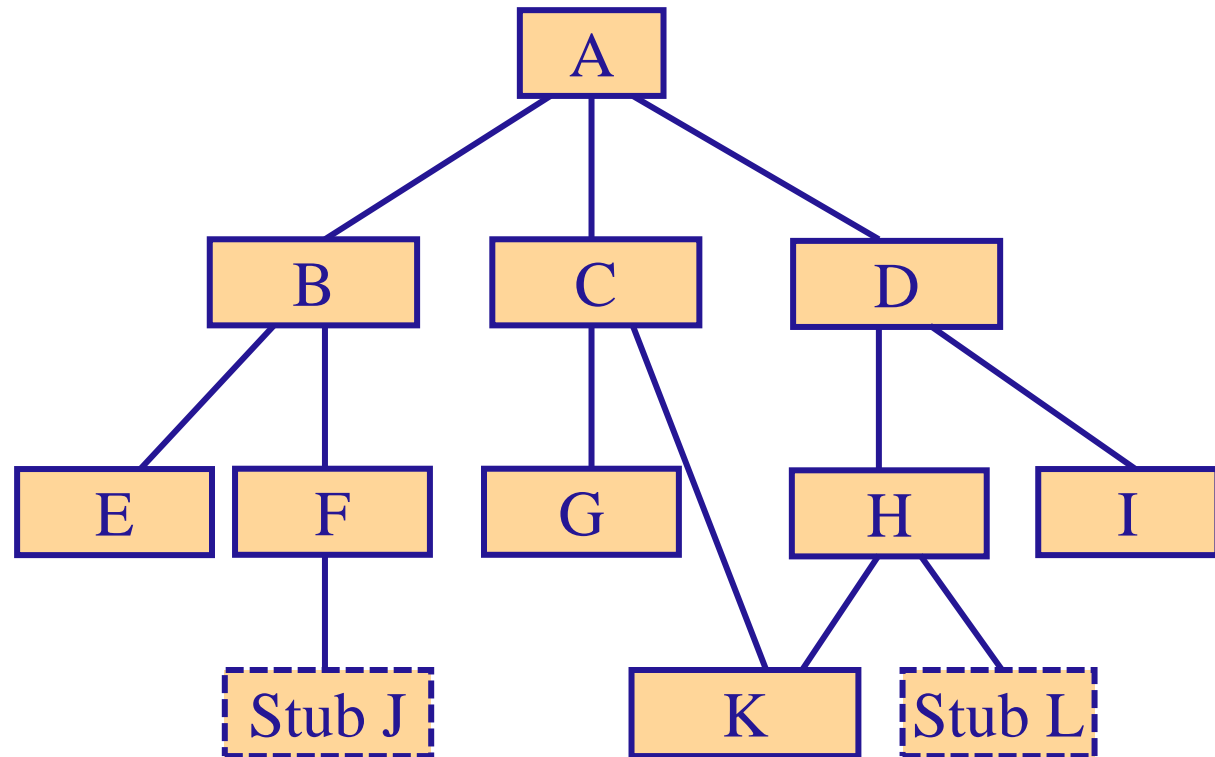
Top-Down Testing



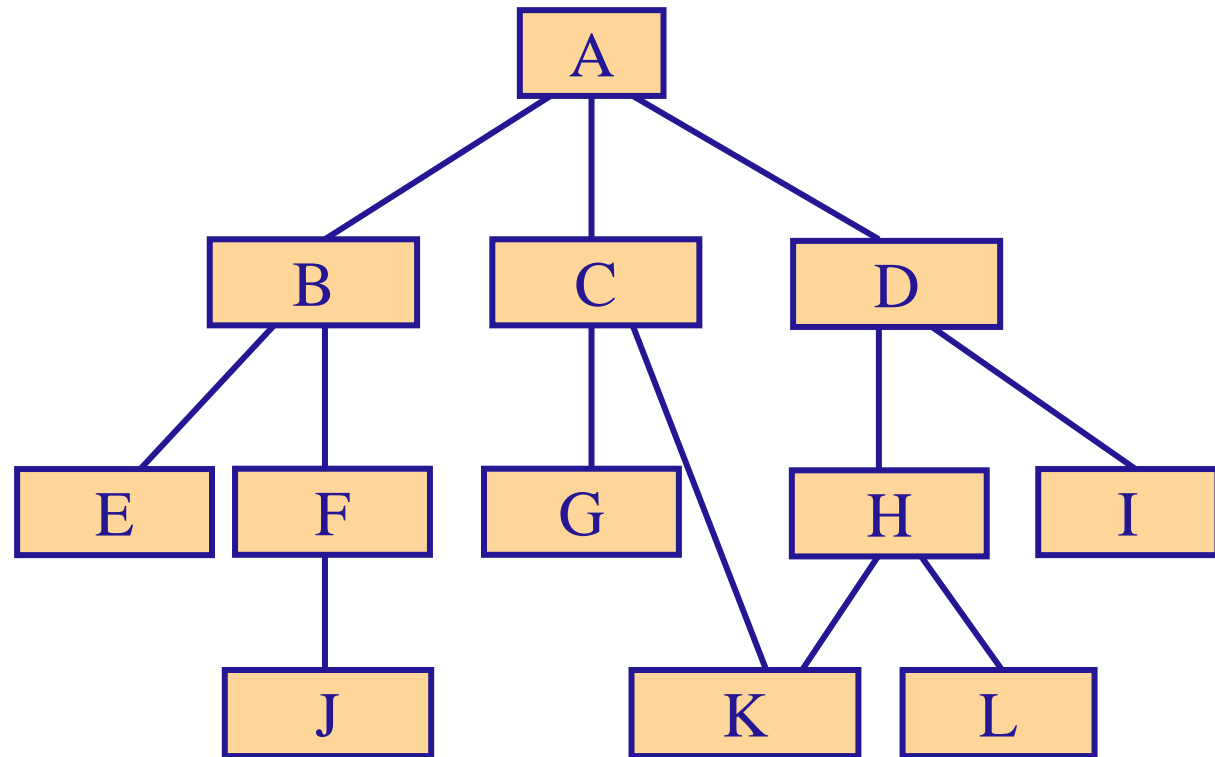
Top-Down Testing



Top-Down Testing



Top-Down Testing





Incremental Testing

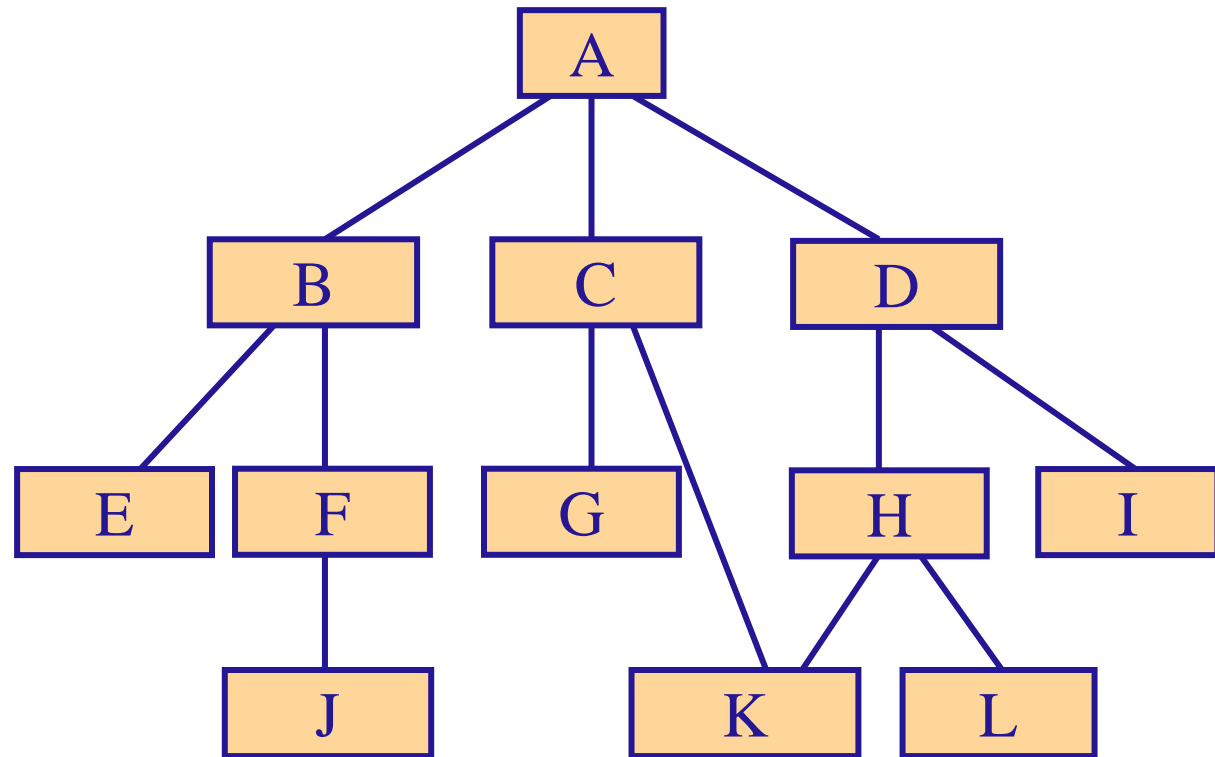




Bottom-Up Testing

- The bottom-up testing starts from the leaf nodes that are the terminal of a program. Before testing, all the modules that are called by the tested unit have been tested.
- To test the lower unit, the driver modules are established.

Bottom-Up Testing



Bottom-Up Testing

E

G

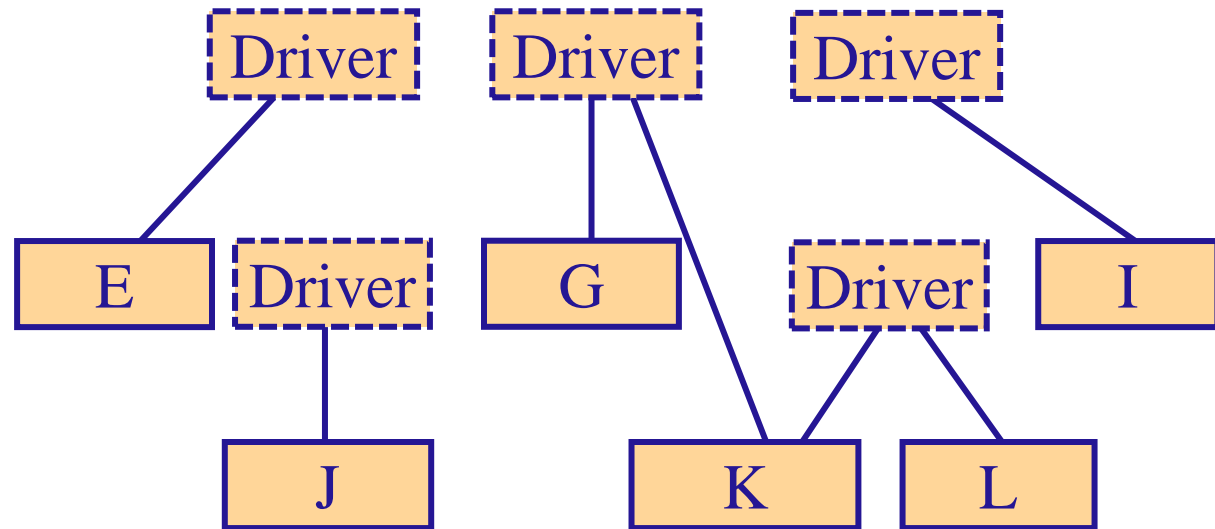
I

J

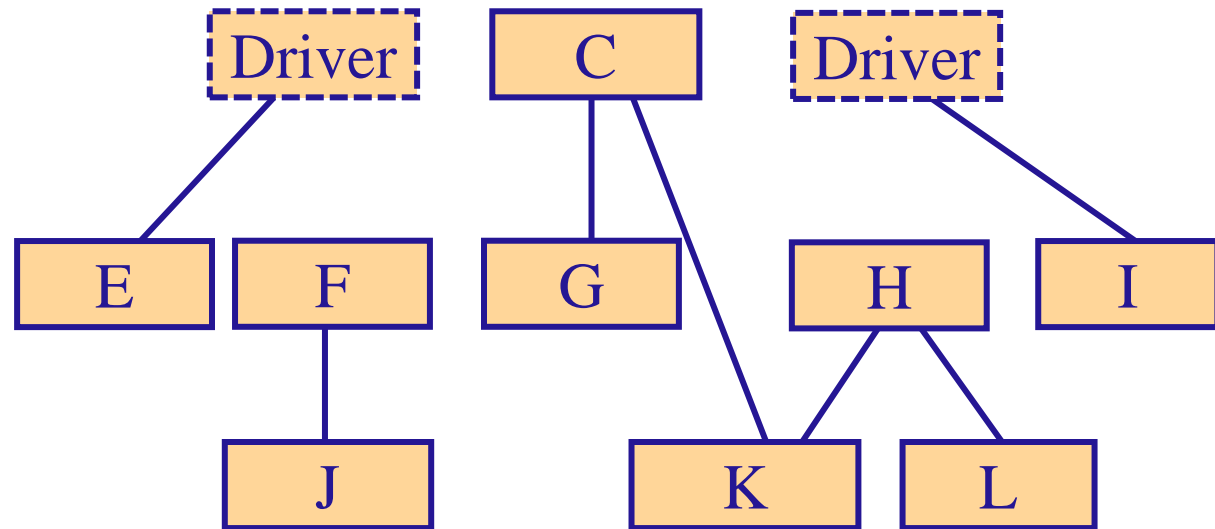
K

L

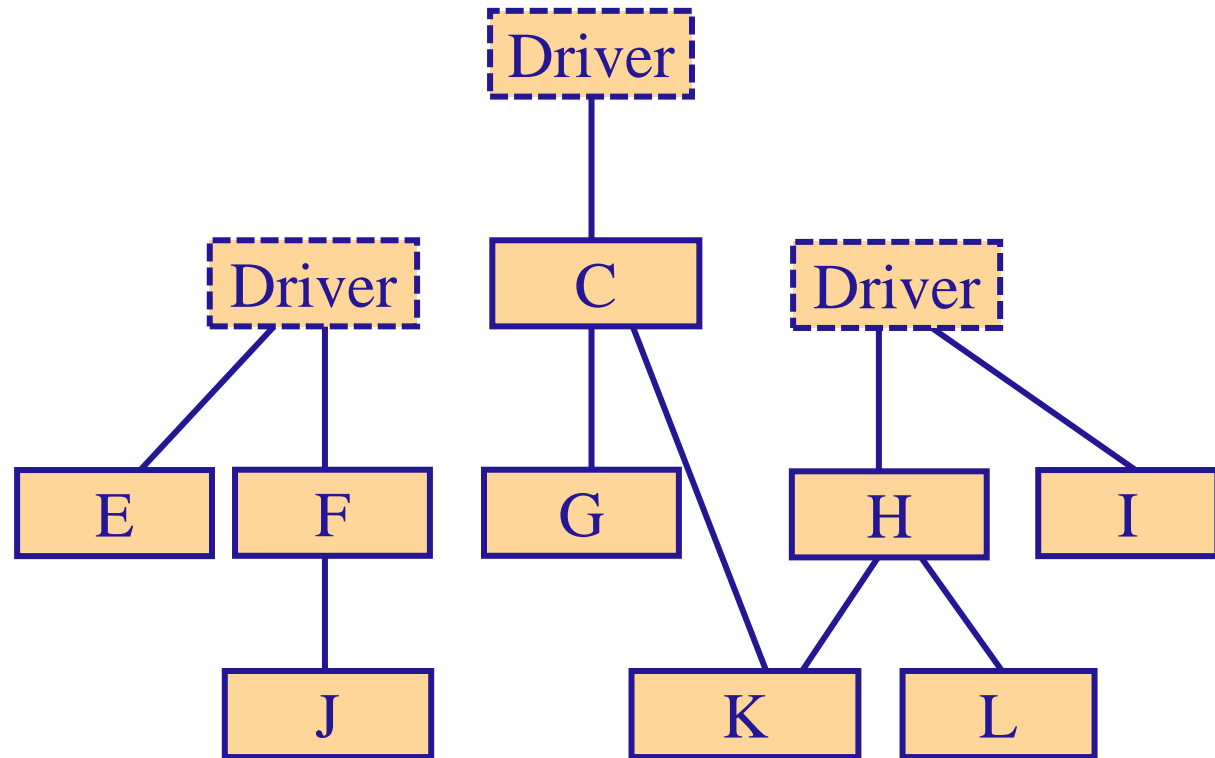
Bottom-Up Testing



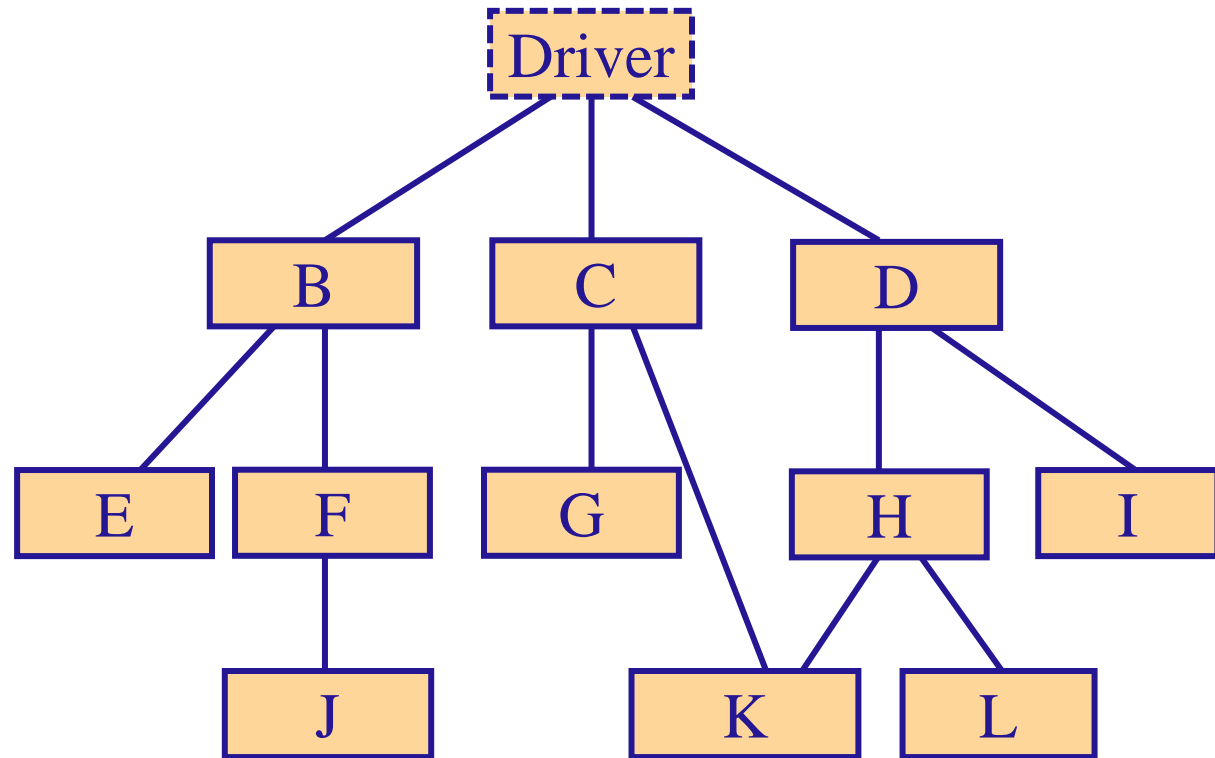
Bottom-Up Testing



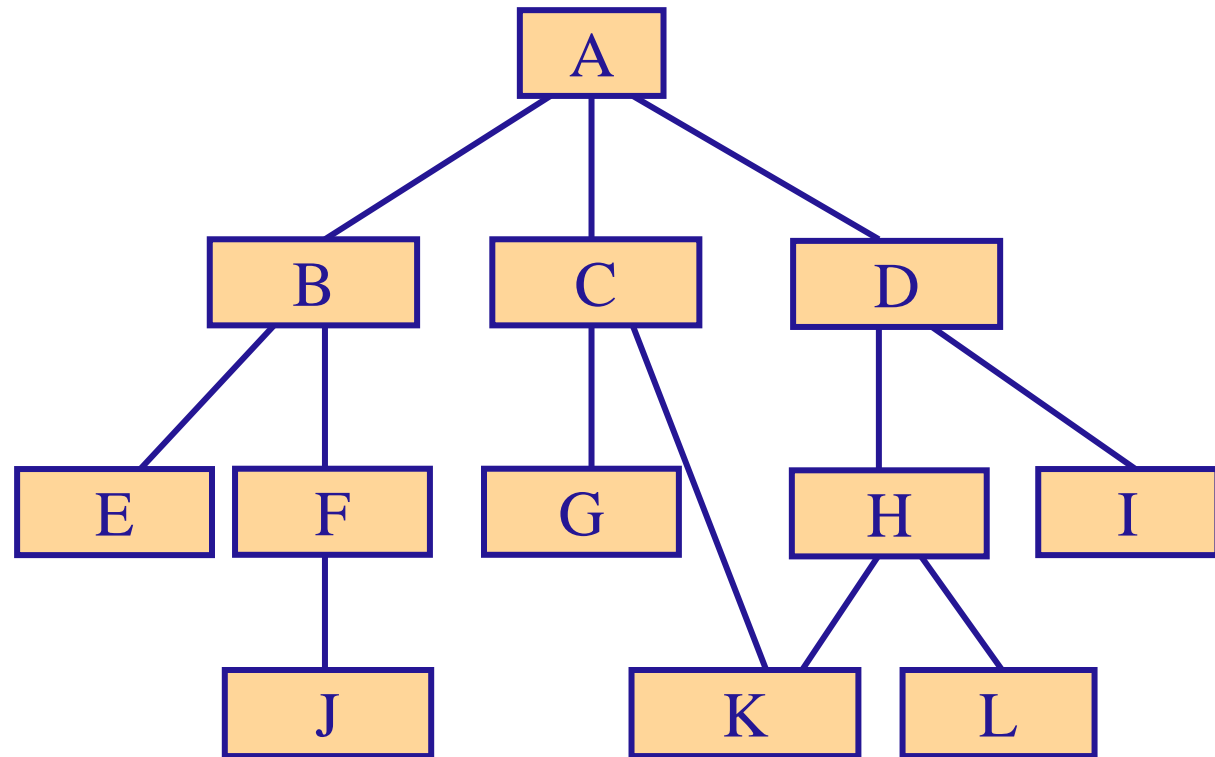
Bottom-Up Testing



Bottom-Up Testing



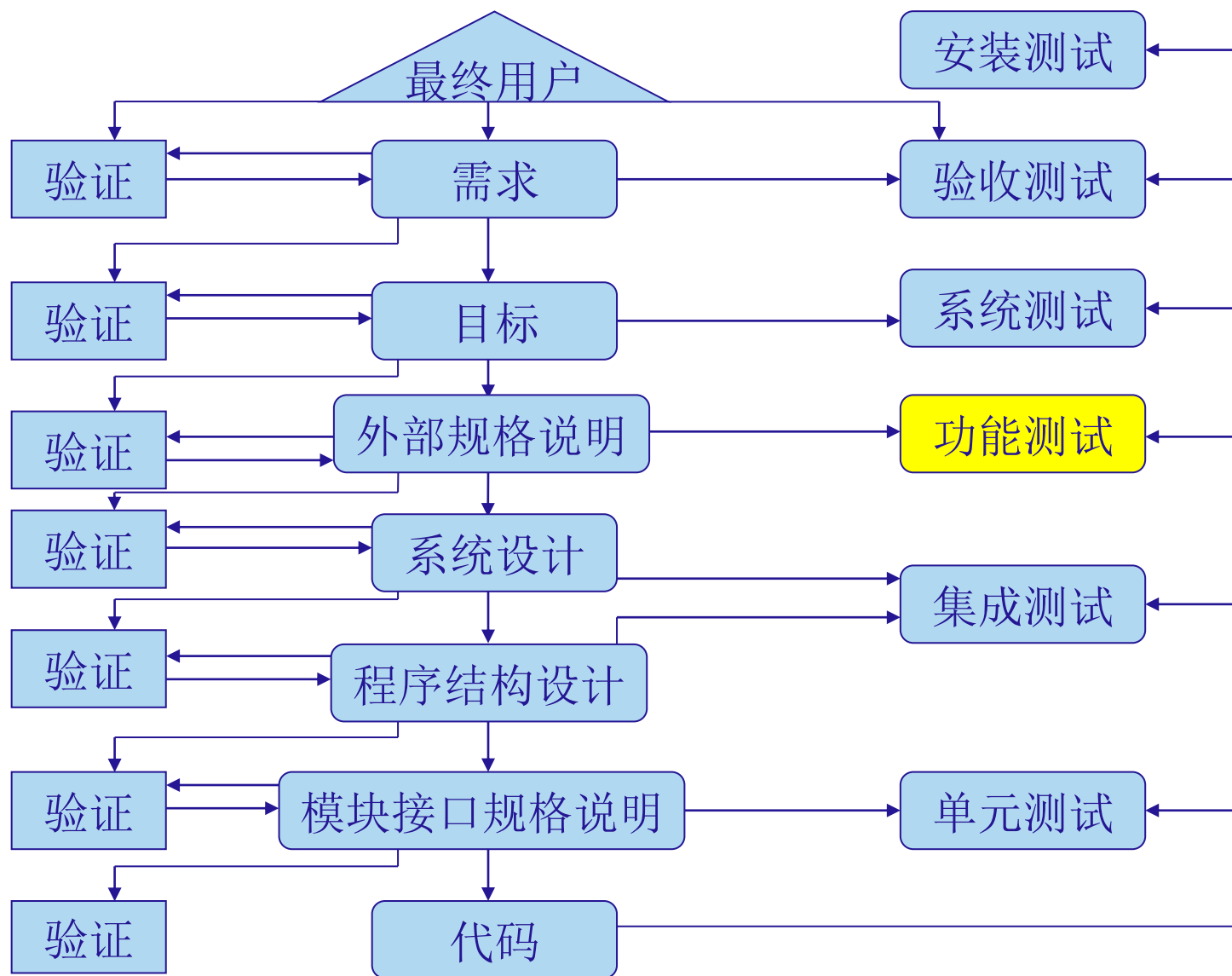
Bottom-Up Testing





Comparison

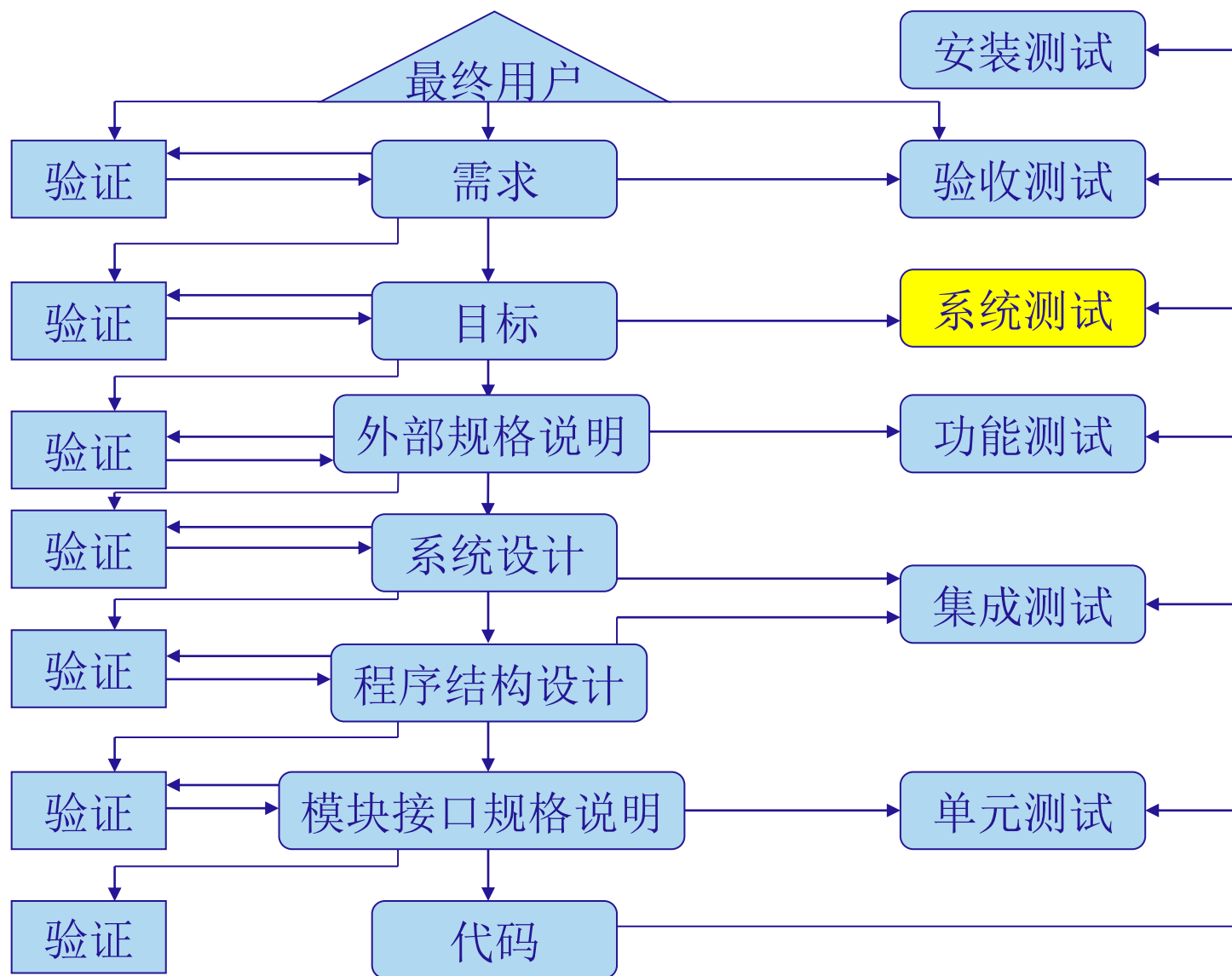
自顶 向下	优点	1、若主要缺陷出现于顶层则非常有利 2、预知框架结构，因此能提早发现主要控制问题
	缺点	1、必须开发桩模块，可能它们比最初表现更复杂 2、创建测试环境可能很难，甚至无法实现 3、观测测试输出比较困难
自底 向上	优点	1、若主要的缺陷发生在程序的底层将非常有利 2、提早发现程序当中的主要算法问题 3、测试环境比较容易建立 4、观测测试输出比较容易
	缺点	1、必须开发驱动模块 2、直到最后一个模块添加进去，程序才形成整体





Functional Testing

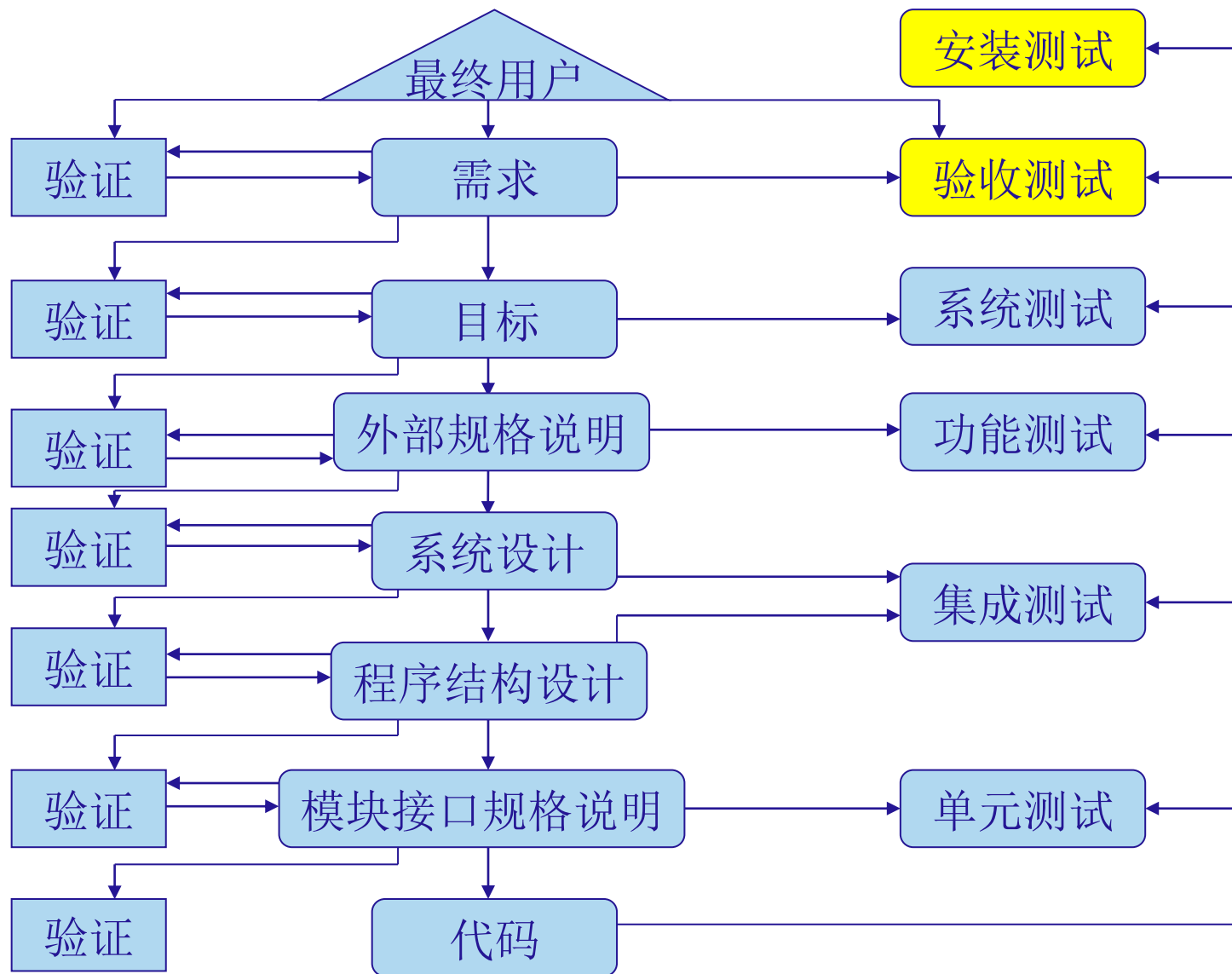
- In the functional testing, one tries to find out the difference between the actual function of the program and its specification. Upon the analysis of the specification, one will get a set of test cases.
- Except for some very small programs, the black-box testing techniques are usually applied, and the techniques, such as equivalence partitioning, boundary value analysis, decision table and cause-effect diagram, are especially applicable here.





System Testing

- System testing, whose aim is to compare the entire system with the initial goals, is a process to find out whether the system is not capable of meeting the goals. So without these goals, one can never executing the system testing.
- It is worth noting that the system testing is not to test the function of the system because it is not based on the specification. Actually the system testing can be viewed as a non-functional testing.
- It is crucial to determine who will be in charge of system testing because the testers here must know customers' attitude and the circumstance for using. Those who develop the software can not act as the system tester.





Acceptance and Installation Testing

- The acceptance testing is the process to compare the program with the initial requirement and the customers' final requirement. It is process executed by customers or the end users, so it is not the duty of those who developed the software.
- The aim of installation testing is not to find the error in the software, but to find the error in the installation process in which many options should be determined by the users and valid configurations should be involved.



Test Plan and Control

- 目标：定义每个测试阶段的目标
- 结束准则：制订准则规定每个测试阶段何时可以结束
- 进度：每个阶段的时间表，何时设计、编写和执行测试用例
- 责任：每个阶段应确定由谁负责设计、编写和执行测试用例，谁负责修改发现的软件错误
- 测试用例库及标准
- 工具：需要使用的测试工具，何时、如何使用



Test Plan and Control

- 计算机时间：每个测试阶段所需的计算机时间
- 硬件配置：描述需要的特别的硬件配置或设备
- 集成：定义程序如何组装在一起，如自顶向下的增量测试
- 跟踪步骤、调试步骤
- 回归测试：对程序作了功能改进或修改之后进行回归测试，目的是判断程序的改动是否引起了程序其他方面的退步。通常是重新执行测试用例集的某个子集。



Ending Criteria for Testing

❖ Valid Criteria

1. 根据特定的测试用例技术来定义准则。例如规定通过了某些来源的所有测试用例后结束。
2. 以确切的数量来描述结束测试的条件。需要预测一些数量：程序错误总数，可能发现的错误比例，错误产生和被发现的阶段。某些条件也可以使用经验数据。
3. 在测试过程中记录每单位时间内发现的错误数量，通过检查统计曲线的形状，决定是继续该阶段的测试还是结束它并开始下个测试阶段。



Ending Criteria for Testing

❖ Invalid Criteria (that should be avoided)

1. 测试时间超过了最初期待的时间长度。
2. 执行完了所有的测试用例后都没有发现错误。

❖ 以上两者在实际应用过程中是经常会犯的问题，原因就在于测试者进行软件测试的动机不是为了发现真正的问题，而是为了快点完成任务。



Common Issues System Testing

- Ability Testing (能力测试)
- Load Testing (容量测试)
- Stress Testing (压力测试)
- Performance Testing (性能测试)
- Memory Testing (存储测试)



Common Issues System Testing

- Ability Testing (能力测试)
- Load Testing (容量测试)
- Stress Testing (压力测试)
- Performance Testing (性能测试)
- Memory Testing (存储测试)



Ability Testing

- The ability testing aims to check whether every ability item has been realized or not.
- The ability testing is not the functional testing, which focuses on whether the software can realize the corresponding functions. The ability testing focuses more on the specific quality to realize a function, such as stability or compatibility.



Common Issues System Testing

- Ability Testing (能力测试)
- **Load Testing (容量测试)**
- **Stress Testing (压力测试)**
- Performance Testing (性能测试)
- Memory Testing (存储测试)



Load Testing & Stress Testing

- The load testing makes the system experience large amount of data, aiming to check whether the software can cope with the load as demonstrated. It will occupy much source. For example, the popular artificial neural networks need load testing when they are run in a computer.
- The stress testing aims to test whether the system can handle high stress, which means that in a short time the data or operations reach their maximum. For example, the actual power source supply will consider the maximum power transfer, so the stress testing is indispensable.



Common Issues System Testing

- Ability Testing (能力测试)
- Load Testing (容量测试)
- Stress Testing (压力测试)
- Performance Testing (性能测试)
- Memory Testing (存储测试)



Performance & Memory Testing

- Performance testing corresponds to the specific goal, such as the accuracy, operating speed, etc. Its aim is to show whether the system can reach its specified performance.
- The memory testing is to check the hardware where the software will be run, such as the storage of the main memory, the hit rate of cache and the structure of virtual memory.



Specific Issues in System Testing

- Configuration Testing (配置测试)
- Compatibility Testing (兼容性测试)
- Foreign Language Testing (外国语言测试)
- Usability Testing (易用性测试)
- Documentation Testing (文档测试)
- Safety Testing (安全性测试)



Specific Issues in System Testing

- Configuration Testing (配置测试)
- Compatibility Testing (兼容性测试)
- Foreign Language Testing (外国语言测试)
- Usability Testing (易用性测试)
- Documentation Testing (文档测试)
- Safety Testing (安全性测试)



Configuration Testing

- ❖ A process for various hardware to run a software.
- ❖ 在家用和商业领域基于标准Windows的PC机的配置可能性：
 - 个人计算机
 - 部件
 - 外设
 - 接口
 - 可选项和内存
 - 设备驱动程序



Configuration Testing

- ❖ 配置测试是必不可少的，原因是硬件的生产厂商并没有执行严格的标准，有时候仅执行松散规范，这样导致软件使用某种硬件配置无法正常工作。
- ❖ 如果准备开始软件的配置测试，就要考虑哪些配置与程序密切相关。
 - 游戏软件要多注意视频和声音部分
 - 贺卡程序要多注意打印问题
 - 传真或通信程序要在多种调制解调器和网络配置下测试



Configuration Defect

- ❖ 判断缺陷是由配置问题产生而不仅仅是一个普通缺陷，最可靠的方法是：在另外一台由完全不同硬件配置的机器上执行导致问题的相同操作。
- ❖ 缺陷来自哪里？
 - 软件包含在多种配置中都会出现的缺陷
 - 硬件设备或者其设备驱动程序可能包含仅由被测软件揭示的缺陷
 - 客户只是要求他们买的软件在自己的配置中正常工作



Workload in Configuration Testing

- ❖ 配置测试工作量可能非常巨大。
- ❖ 例如：测试运行于Windows的新游戏。该游戏画面丰富，具有多种音效，允许多个用户通过电话线对抗，而且可以打印游戏细节以便进行策划。那么至少要考虑各种显卡、声卡、调制解调器和打印机进行配置测试。
- ❖ 如果市场上大致有300种显卡、200种声卡、1000种调制解调器、1000种打印机。测试组合的数目总共约 6×10^{10} 种。



Ways to Reduce the Workload

- ❖ 减少工作量的方法是等价类划分。可以参考依据以下几个标准来寻找等价类。
 - 流行程度
 - 年代
 - 类型
 - 生产厂商
- ❖ 等价类划分存在一定的风险，因为无法做到完全测试。



Executing Process

- ❖ 确定所需的硬件类型。
- ❖ 确定有哪些厂商的硬件、型号和驱动程序可用。
- ❖ 确定可能的硬件特性、模式和选项。
- ❖ 将确定后的硬件配置缩减为可控制的范围，如使用等价类划分法。
- ❖ 明确与硬件配置有关的软件唯一特性，弄清楚软件测试的哪些功能与硬件配置相关。
- ❖ 设计在每一种配置中执行的测试用例。
- ❖ 在每种配置中执行测试。
- ❖ 反复测试直到小组对结果满意为止，即达到没有未解决的缺陷或缺陷限于不常见或不可能的配置上。



How to Gain the Hardware

- ❖ 只买可以或者将会经常使用的配置。
- ❖ 与硬件生产厂商联系，看他们是否能够租借甚至赠送某些硬件。
- ❖ 向公司的员工询问他们能够提供什么硬件。
- ❖ 如果预算充足，拿到专业配置和兼容性测试实验室联系进行外协测试。



THANK YOU!