

Beijing Jiaotong University

2020—2021 School Year Second Semester Exam (A)

Course Name: Software Quality Assurance and Testing Techniques

Teacher: Xiaoping Che, Qian Zhang, Haiming Liu Major: _____

Class: _____ Name: _____ Student ID: _____

No.	1	2	3	4	5	6	Total Score
Score							
Examiner							

Part 1. Choose the correct answer from following choices. (2×10 marks)

- D 1. Which of the following names is **not** a common name of a Bug?
- A. Fault
 - B. Error
 - C. Failure
 - D. OMGx
- A 2. Which of the following testing method **cannot** be put in the same category with the others (in the perspective of verification and validation)?
- A. Acceptance Testing - spec
 - B. System Testing
 - C. Integration Testing
 - D. Unit Testing
- B 3. Which of the following aspects does **not** belongs to the definition of Non-Functional aspects?
- A. Performance
 - B. Correctness
 - C. Usability
 - D. Security

4. Which of the following is **not** included in the Specification attribute checklist?
- A. Completeness
 - B. Testability ?
 - C. Intuitive
 - D. Code/Design-free!
5. Which of the following is a correct description of testing?
- A. Testing is the process of demonstrating that errors are not present.
 - B. The purpose of testing is to show that a program performs its intended functions correctly.
 - C. Testing is the process of executing a program with the intent of finding errors.
 - D. Testing is the process of establishing confidence that a program does what it is supposed to do.
6. Which testing method is **not** in the Control-flow Testing Criteria?
- A. Iteration Testing
 - B. Statement Testing ✓
 - C. Branch Testing ✓
 - D. Path Testing ✓
7. A Good UI design should **not** be?
- A. Consistent ✓
 - B. Comfortable ✓
 - C. Flexible ✓
 - D. Miscellaneous ✓
8. In the code review part, which of the following error **cannot** be detected?
- A. Data Reference Errors
 - B. Memory usage Errors ?
 - C. Computation Errors
 - D. Data Declaration Errors
9. Which of the following methods **cannot** be used to describe software specification?
- A. SDL
 - B. GUI
 - C. MSC
 - D. Z NOTATION

- C 10. Which of the following is **not** included in the Software Quality Assurance?
- A. SQP *?plan*
 - B. SQC *control*
 - C. SQV
 - D. SQM *management*

Part 2. The following descriptions are True or False? (2×5 marks)

11. It is possible to test a program completely. *X* True or False
12. Testing can show the absence of bugs. *X* True or False
13. Not all bugs found will be fixed. *✓* True or False
14. It is difficult to say when a bug is indeed a bug. *✓* True or False
15. In white box testing, “Exercise every statement at least once” equals to “Exercise every branch (in each direction) at least once” *X* True or False

Part 3. Please fill the correct answer in the blanks. (2×5 marks)

16. In the Software Development V model, it contains Requirement Analysis, System Design, Program Design, Unit Testing, Function Testing, System Testing, Acceptance, and Operation Maintenance.

17. Please provide the full name of abbreviations SRS software requirements specification, PDL program development language in software testing domain.

18. There are two strategies in integration testing: Bottom-up Testing and Top-down Testing.

19. In unit testing, stub represents a component that calls component to be tested.

20. In Beta testing, rather than test team member, development team member, customer will involve into the testing process.

Part 4. Short questions. (20 marks)

21. Please explain the definition of Equivalence Partitioning and Boundary Value Analysis, and their different impacts on designing test cases. (10 marks)

22. Please explain the definition of Validation and Verification, and explain the difference between them. (10 marks)

Part 5. Long Questions. (2*20 marks)

23. The schedule manager schedules a meeting to (a) time slot(s) as part of the 5-day schedule for the event. To accomplish this, the user must input three items:

1. The day (1, 2, 3, 4, or 5), chosen from a drop-down list.
2. The time slot(s), chosen through some type of user interface. Each day of the event lasts from 10am-6pm. There are 8, 1-hour time slots available per day (10-11am, 11am-12pm, 12-1pm, 1-2pm, 2-3pm, 3-4pm, 4-5pm, 5-6pm). A meeting can be scheduled for a 1-hour, 2-hour, or 3-hour time slot.
3. The name of the meeting (chosen from a drop-down list of meetings that the system already knows about)

The output would be a meeting being scheduled for a particular date and time (as reflected in the schedule). For example, if one of my test cases consisted of: {Day 3, 11am-1pm, The Go-Go's}, the expected output for this test case would be the Go-Go's being scheduled for Day3 of the festival in the 11am-1pm time slot.

(The minimum time unit is 1 hour. Minutes are not considered in this question)

Please provide the **Equivalence Partitioning** and **Boundary Value Analysis** of this use case and provide relevant **Test Cases Design**.

Test Case Example:

Test Case No.	Input Value 1	Input Value 2	Input Value 3	Expected Output	Description
1					

24. **(20 marks)** Now we have two code written by some programmers which are shown below, please provide the Control Flow Testing process of each code, including **Control Flow Graph, Cyclomatic complexity and Basis Set. And you can report the bugs if you find any.**

Code 1: **(8 marks)**

```

1. function radixSort(arr, maxDigit) {
2.     varmod = 10;
3.     vardev = 1;
4.     for(vari = 0; i < maxDigit; i++, dev *= 10, mod *= 10) {
5.         for(varj = 0; j < arr.length; j++) {
6.             varbucket = parseInt((arr[j] % mod) / dev);
7.             if(counter[bucket]==null) {
8.                 counter[bucket] = [];
9.             }
10.            counter[bucket].push(arr[j]);
11.        }
12.        varpos = 0;
13.        for(varj = 0; j < counter.length; j++) {
14.            varvalue =null;
15.            if(counter[j]!=null) {
16.                while((value = counter[j].shift()) !=null) {
17.                    arr[pos++] = value;
18.                }
19.            }
20.        }
21.    }
22.    return arr;
23. }
```

Code 2: (12 marks)

```
1. public class TestBreakContinue {
2.     public static void main(String[] args) {
3.         int total = 0;
4.         System.out.println("Begin");
5.         while (true) {
6.             total++;
7.             int i = (int) Math.round(100 * Math.random());
8.             if (i == 88) {
9.                 break;
10.            }
11.        }
12.        for (int i = 100; i < 150; i++) {
13.            if (i % 3 == 0) {
14.                continue;
15.            }
16.            System.out.println(i);
17.        }
18.        int count = 0;
19.        outer: for (int i = 101; i < 150; i++) {
20.            for (int j = 2; j < i / 2; j++) {
21.                if (i % j == 0)
22.                    continue outer;
23.            }
24.            System.out.print(i + " ");
25.        }
26.    }
27. }
```

Beijing Jiaotong University

2021—2022 School Year Second Semester Exam (A)

Course Name: Software Quality Assurance and Testing Techniques

Teacher: Xiaoping Che, Haiming Liu, Haonan Tong Major: _____

Class: _____ Name: _____ Student ID: _____

No.	1	2	3	Total Score
Score				
Examiner				

Submit Link: 【腾讯文档】软件质量保证与测试技术(che)

<https://docs.qq.com/form/page/DSGIwaW1PbmtqTHI2>

Handwritten notes: f 0:00, s 9:30, f 21:30, s 26:00. A timeline diagram shows intervals: 0:00 to 9:30, 9:30 to 16:00, 16:00 to 21:30, and 21:30 to 26:00.

Part 1. Black Box Testing and Test cases Design. (5*6=30 marks)

If you take the train before 9:30 AM or in the afternoon after 4:00 PM until 7:30 PM ('rush hour') you must pay full fare. A saver ticket is available for trains between 9:30 AM and 4:00 PM, and after 9:30 PM.

Q1: What are the partitions and boundary values to test the train times for this ticket types?

Q2: Which are valid partitions and which are invalid partitions?

Q3: What are the boundary values? (A table may be useful)

Q4: Design test cases for the partitions and boundaries.

Q5: Do you have any questions about this 'requirement'? Is anything unclear?

Handwritten notes for Q1 and Q2:

Q1 EP (Effective Partitions): [0:00, 9:30), [9:30, 16:00], [16:00, 21:30], [21:30, 23:59]

BV (Boundary Values): 0:00, 0:01, 9:00, 9:29, 9:30, 9:31, 13:00, 15:59, 16:00, 16:01, 21:00, 21:29, 21:30, 21:31, 23:00, 23:58, 23:59

Q2 Valid [0:00, 23:59], Invalid [Min, 0:00), (23:59, Max]

Q3?

Q4 take train time

excepted Outcomes

0:00

0:01

9:00

9:29

9:30

9:31

13:00

15:59

16:00

16:01

21:00

21:29

21:30

21:31

23:00

23:58

23:59

full fare

saver ticket

full fare

saver ticket

Q5 full fare before 9:30 A.M.

和 saver ticket after 9:30 P.M. 会重复, 应该怎么划分

Part 2. White Box Testing and Test cases Design. (20*2=40 marks)

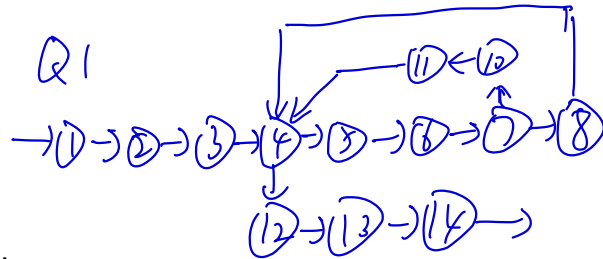
Now we have two code written by some programmers which are shown below, please provide the Control Flow Testing process of each code, including **Control Flow Graph**, **Cyclomatic complexity** and **Basis Set**. And you can report the bugs if you find any.

Q1:

```

1  int main() {
2      int i;
3      double number, sum = 0.0;
4      for (i = 1; i <= 10; ++i) {
5          printf("Enter a n%d: ", i);
6          scanf("%lf", &number);
7          if (number < 0.0) {
8              continue;
9          }
10         sum += number;
11     }
12     printf("Sum = %.2lf", sum);
13     return 0;
14 }

```



Cyclomatic complexity = ?

Basis Set: 1. 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 → 9 → 10 → 11 → 12 → 13 → 14

2. 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 → 9 → 10 → 11 → 12 → 13 → 14

3. 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 → 9 → 10 → 11 → 12 → 13 → 14

bugs: 1. 不可交换性

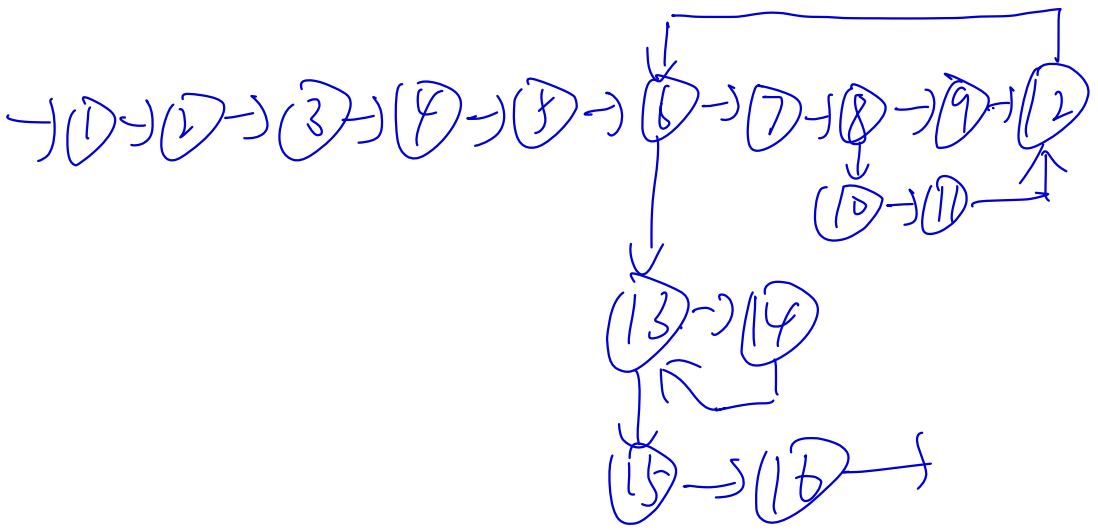
4. 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 → 9 → 10 → 11 → 12 → 13 → 14

Q2:

```

1  public static void main(String[] args) {
2      File f = new File("ciaFactBook2008.txt");
3      Scanner sc;
4      sc = new Scanner(f);
5      Map<String,Integer>wordCount=newTreeMap<String,Integer>();
6      while(sc.hasNext()) {
7          String word = sc.next();
8          if(!wordCount.containsKey(word))
9              wordCount.put(word, 1);
10         else
11             wordCount.put(word, wordCount.get(word) + 1);
12     }
13     for(String word : wordCount.keySet())
14         System.out.println(word + " " + wordCount.get(word));
15     System.out.println(wordCount.size());
16 }

```



Cyclomatic complexity: 4

Basic Set:

1, 2, 3, 4, 5, 6, 13, 15, 16

1, 2, 3, 4, 5, 6, 13, 14, 13, 15, 16

1, 2, 3, 4, 5, 6, 7, 8, 9, 12, 6, 13, 15, 16

1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 6, 13, 15, 16

bugs: 文件读取失败, 除 17-8 以外编码?

Part 3. Understanding of Testing. (5*6=30 marks)

Q1: Please put the test cases that implement the following test conditions into **the best order** for the test execution schedule and **explain why**, for a test that is checking modifications of customers on a database.

- 1) Print modified customer record
- 2) Change customer address: House number and street name
- 3) Capture and print the on-screen error message
- 4) Change customer address: Postal code
- 5) Confirm existing customer is on the database by opening that record
- 6) Close the customer record and close the database
- 7) Try to add a new customer with no details at all

5, 2, 4, 1, 3, 3, 6

Q2: Why are **both behavioral testing** (specification-based testing) and **structural testing** (structure-based testing) techniques useful?

行为测试输入与输出是否符合预期
结构测试代码逻辑

Q3: What are the key characteristics of **structural testing** (structure-based testing) techniques?

white box testing, code coverage, defect based, Test case design

Q4: What are the differences of **Unit Testing**, **Integration Testing**, **System Testing**, and **Acceptance Testing**?

Q5: What are the differences of **Software Quality Assurance** and **Software Quality Control**?