

下六个月软件评测师考试真题及答案-下午卷

试题一

【阐明】

逻辑覆盖法是设计白盒测试用例的重要措施之一,通过对程序逻辑构造的遍历实现程序的覆盖。针对如下由C语言编写的程序,按规定回答问题。

```
int XOR (char * filename, unsigned long key) {  
    FILE * input = NULL, *output = NULL;          //1  
    char * outfilename = NULL;  
    int len = strlen (filename) ;  
    unsigned char buffer;  
    if ( (filename[len-2] == '.') && (filename[len-1] == 'c')) {    //2,3  
        outfilename = new char[len+1];                //4  
        strcpy (outfilename,filename) ; outfilename[len-2] = '\0';  
    }  
    else{                                              //5  
        outfilename = new char[len+5];  
        strcpy (outfilename , filename) ;  
        strncat (outfilename, ".c", 2) ;  
    }  
  
    input = fopen (filename," rb") ;  
    if ( input = NULL) {                                //6  
        cout<<"Error opening file " << filename <<endl ;    //7  
        delete [] outfilename;  
        outfilename = NULL;  
        return 1;  
    }  
    output = fopen (outfilename, " 'wbf") ;  
    if ( output = NULL) {                                //8
```

```

        cout<< "Error creating output file " << outfile_name <<endl;           //9
        delete [] outfile_name;
        outfile_name = NULL;
        return 1;
    }

    while (! feof (input)) {                                                    //10
        if ( fread (&buffer,sizeof (unsigned char) , 1,input) != 1 ) {        //11
            if ( ! feof (input)) {                                              //12
                delete [] outfile_name;                                         //13
                outfile_name = NULL;
                fclose (input) ;
                fclose (output) ;
                return 1;
            }
        }
        else{                                                                    //14
            buffer A= key;
            fwrite (&buffer,sizeof (unsigned char) , 1,output) ;
        }
    }

    fclose (input) ;                                                            //15
    fclose (output) ;
    delete [] outfile_name;
    return 0;
}

```

【问题 1】(6 分)

請給出满足 100%DC（鉴定覆盖）所需的逻辑条件。

【问题 2】(10 分)

請画出上述程序的控制流图，并计算其控制流图的环路复杂度 $V(G)$ 。

【问题 3】（4 分）

請給出問題 2 中控制流图的线性无关途径。

答案:

【问题 1】

`(filename[len-2] == '.') && (filename[len-1] == 'c')` 真, 假

`input = NULL` 真, 假

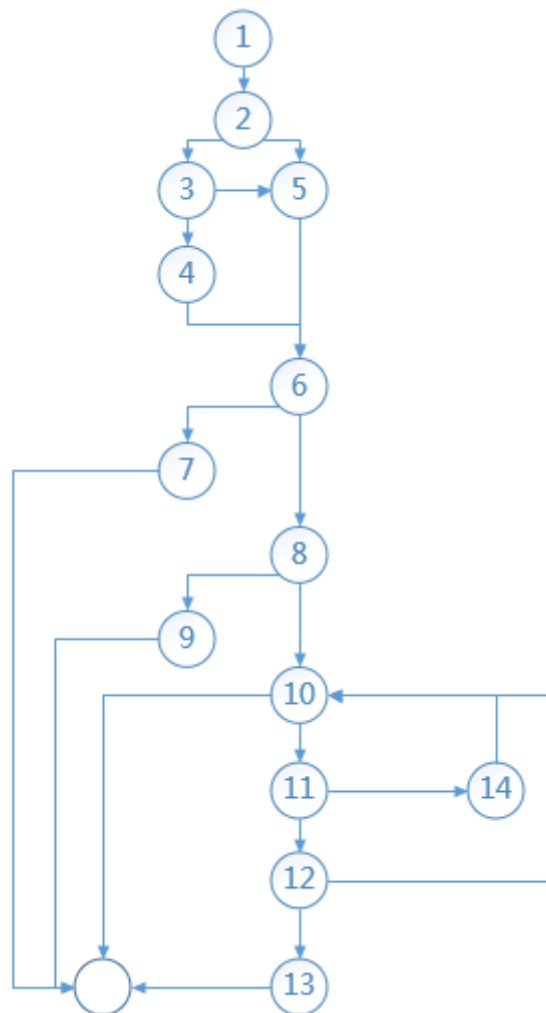
`output = NULL` 真, 假

`!feof(input)` 真, 假

`fread(&buffer, sizeof(unsigned char), 1, input) != 1` 真, 假

`!feof(input)` 真, 假

【问题 2】



$V(G) = 8$

【问题 3】

1、1 2 3 4 6 7

2、1 2 3 5 6 7

3、1 2 5 6 7

4、1 2 5 6 8 9

5、1 2 5 6 8 10

6、1 2 5 6 8 10 11 12 13

7、1 2 5 6 8 10 11 14

8、1 2 5 6 8 10 11 12

或者其他符合规定的等价的途径组合

解析

【问题 1】

鉴定覆盖：设计足够的测试用例，使得程序中的每个鉴定至少都获得一次“真值”或“假值”，或者说使得程序中的每一种取“真”分支和取“假”分支至少都经历一次

逻辑条件为：

`(filename[len-2] == '.') && (filename[len-1] == 'c')` 真，假

`input = NULL` 真，假

`output = NULL` 真，假

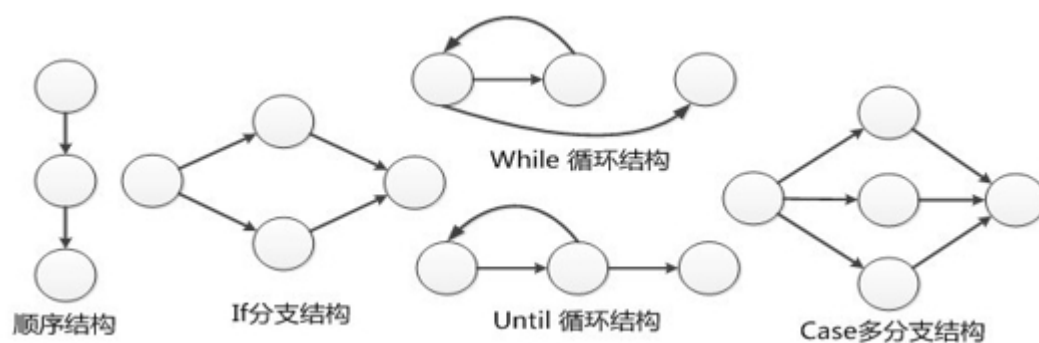
`!feof(input)` 真，假

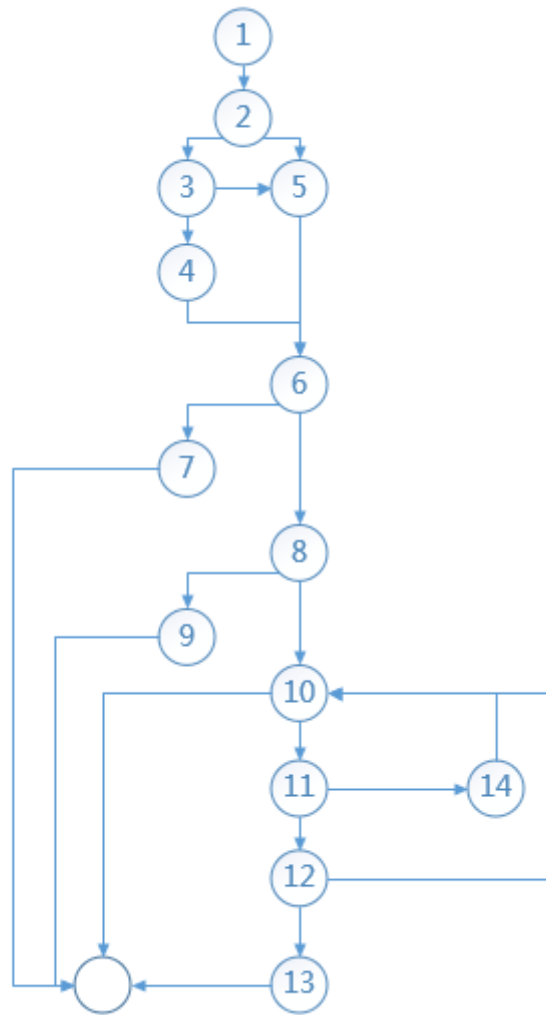
`fread(&buffer,sizeof(unsigned char),1,input) != 1` 真，假

`!feof(input)` 真，假

【问题 2】

控制流图是描述程序控制流的一种图示措施。其基本符号有圆圈和箭头，圆圈为控制流图中的一种结点，表达一种或多种无分支的语句；带箭头的线段称为边或连接，表达控制流。基本构造如下所示：





控制流程图的环路复杂性 $V(G)$ 等于：

- (1) 控制流程图中的区域个数。
- (2) 边数-结点数+2。
- (3) 鉴定数+1。

$$V(G) = 8$$

【问题 3】

线性无关途径是指包括一组此前没有处理的语句或条件的一条途径。从控制流图来看，一条线性无关途径是至少包具有一条在其他线性无关途径中从未有过的边的途径：

- 1、1 2 3 4 6 7
- 2、1 2 3 5 6 7
- 3、1 2 5 6 7
- 4、1 2 5 6 8 9

5、 1 2 5 6 8 10

6、 1 2 5 6 8 10 11 12 13

7、 1 2 5 6 8 10 11 14

8、 1 2 5 6 8 10 11 12

试题二

【阐明】

某软件企业為某银行设计开发了一套网上银行系统，该系统提供基本的支付、查询、转账和信息修改等功能。银行对网上银行系统提出了初步的性能指标：

- (1) 交易响应时间不超过 2s
- (2) 并发顾客数 ≥ 1000
- (3) CPU 运用率不超过 80%
- (4) 系统需要 7*24 小时不间断的稳定运行
- (5) 每秒事务数为 7
- (6) 交易成功率为 100%

现需要对该软件进行性能测试。

【问题 1】(4 分)

常见的性能测试包括负载测试、压力测试、并发性能测试、疲劳强度测试和大数据量测试等。针对题目中所述的 6 个性能指标，并发性能测试和疲劳强度测试所波及的指标各有哪些？

【问题 2】

性能测试中，针对一种单独的性能指标，往往需要采用多种不一样的测试措施。该软件企业需要测试性能指标 (1) 和 (5)，设计了如下的测试方案：

测试常规状况下的并发顾客数，逐渐增长并发顾客数，分别测试：

- (1) 在响应时间为 2s 时，系统所能承受的最大并发访问顾客的数量；
- (2) 系统在多大的并发访问顾客数量下，响应时间不可接受（例如超过 2s）。

请指出这两项测试分别属于哪种类型的测试，并分别解释这两种测试类型的基本概念。

【问题 3】

在测试性能指标 (5) 时，该软件企业在客户端模拟大量并发顾客来执行业务操作，记录平均的每秒事务数。该软件企业认为客户端接受响应信息与该性能指标的测试无关，因此在模拟客户端上发起正常业务申请，接受系统响应后直接丢弃响应信息，没有进行功能校验。请阐明该软件企业的做法对的与否，并简要阐明原因以及执行功能校验的副作用。

答案：

【问题 1】

并发性能测试波及的指标有（1）、（2）、（5）

疲劳强度测试波及的指标有（4）

【问题 2】

第一种测试属于负载测试，第二种属于压力测试。

负载测试是通过逐渐增长系统负载，测试系统性能的变化，并最终确定在满足性能指标的状况下，系统所能承受的最大负载量的测试。

压力测试是通过逐渐增长系统负载，测试系统性能变化，并最终确定在什么负载条件下系统性能处在失效状态，以此来获得系统能提供的最大服务级别的测试。

【问题 3】

该软件企业的做法是错误的。

该软件企业在负载压力测试中没有进行功能校验，忽视了负载压力状况下的功能不稳定性问题。没有对的功能保证，负载压力性能测试就失去了意义。

在测试过程中进行功能校验，需要记录业务操作成果，会导致资源消耗、操作行为增长以及产生大量日志等问题。

解析：

【问题 1】

本题重要考察负载压力测试有关知识及应用。

负载压力测试是指在一定约束条件下测试系统所能承受的并发顾客量、运行时间、数据量等，以确定系统所能承受的最大负载压力。负载压力测试是性能测试的重要构成部分，包括负载测试、压力测试、并发性能测试、疲劳强度测试、大数据量测试等内容。

本问题考察负载压力测试的基本概念。

负载测试是通过逐渐增长系统负载，测试系统性能的变化，并最终确定在满足性能指标的状况下，系统所能承受的最大负载量所进行的测试。

压力测试是通过逐渐增长系统负载，测试系统性能的变化，并最终确定在什么负载条件下，系统性能处在失效状态，以此来获得系统能提供的最大服务级别的测试。因此说，压力测试是一种特定类型的负载测试。

并发性能测试，包括了负载测试和压力测试。通过逐渐增长并发顾客数负载，直到系统的瓶颈或者不能接受的性能点，通过综合分析交易执行指标、系统资源监控指标来确定系统并发性能。并发性能测试是负载压力测试中的重要内容。

疲劳强度测试，一般是采用系统稳定运行状况下可以支持的最大并发顾客数，或者平常运行顾客数，持续执行一段时间业务，保证到达系统疲劳强度需求的业务量，通过综合分析交易执行指标、系统资源监控指标，确定系统处理最大工作量强度 1 生能的过程。一般状况下运用疲劳强度测试老模拟系统平常业务操作。

大数据量测试，包括独立的数据量测试和综合数据量测试。独立的数据量测试是针对某些系统存储、传播、记录、查询等业务进行的大数据量测试；综合数据量测试是指和压力测试、负载性能测试、疲劳强度测试相结合的综合测试。

从题干列举的 6 种性能指标看，根据各类测试的定义，可以得出（1）、（2）、（5）属于并发性能测试，（4）属于疲劳强度测试。（3）属于一种系统资源监控的指标，而（6）的定义不明确，由于交易成功率取决多种原因，包括业务数据、顾客祈求数据、基础软件、网络环境等等。

【问题 2】

本问题考察负载测试和压力测试的区别。

从问题 1 的分析中可以看出，负载测试强调的是在满足性能指标的状况下，系统所能承受的最大负载量的测试，前提是系统正常工作；而压力测试则是要使得系统性能处在失效状态，既压力测试是为了发目前什么条件下系统的性能会变得不可接受。

因此，通过对本问题描述的分析可以得出，“在响应时间为 2s 时，系统所能承受的最大并发访问顾客的数量”属于负载测试，而“系统在多大的并发访问顾客数量下，响应时间不可接受（例如超过 2s）”属于压力测试。

【问题 3】

本问题考察负载压力测试中的功能内容校验。

负载压力测试中的功能内容校验，指的是在进行负载压力测试时，需要核查功能的执行与否正常，在功能异常或错误时，需要记录所产生的错误。假如没有功能执行成果对的的保证，负载压力性能测试失去了意义。对本为题而言，假如不能保证业务操作的对的，则平均每秒事务数的指标是不对的。

执行功能内容校验也会带来某些副作用，包括资源消耗、操作行为增长，以及产生大量日志等问题。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。

如要下载或阅读全文，请访问：

<https://d.book118.com/405021004302011341>