Homework Tutorial 6: Verification: Testing and Analysis

Wenchao GU

csci3100@cse.cuhk.edu.hk

April 19, 2021

Outline

1. Program Testing
2. White-box Testing
3. Black-box Testing
4. Code Coverage
5. Stub and Driver Modules

1. Program Testing

Source: 2019 / Homework 6 / Question 1

Program Testing\Question 1

Given the following Java program:

```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11. if( s.charAt(i) == '0')
12.     result = -result;
13. if( result>2147483648 || result<-2147483648)
14.     throw new Exception();
15. return result;
16.}
```

Draw the control flow graph of this program. (You can lump sequential statements into one node. The procedure partition is treated as one node without further analysis.).

Program Testing\Answer 1

Given the following Java program:

```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Program Testing\Answer 1

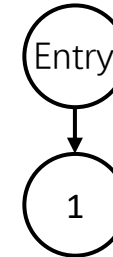Given the following Java program:

Entry

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Draw the entry node at first!

Program Testing\Answer 1

Given the following Java program:
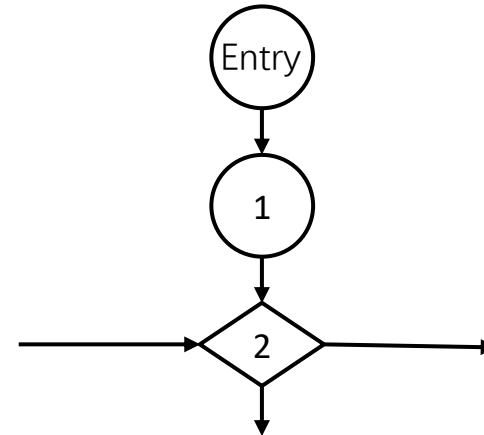
```java
1.     public int Transform(String s) throws Exception{
2.         int result = 0;
3.         int power = 0;
4.       for (int i = s.length() - 1; i >= 1; i--){
5.             if(s.charAt(i) == '1')
6.                 result += Math.pow(2,power);
7.             else if(s.charAt(i) != '0')
8.                     throw new Exception();
9.             power ++;
10.     }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Entry

→

1

Program Testing\Answer 1

Given the following Java program:
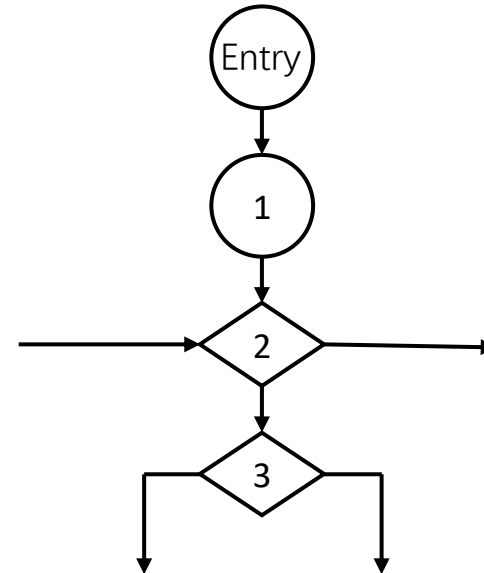
```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:
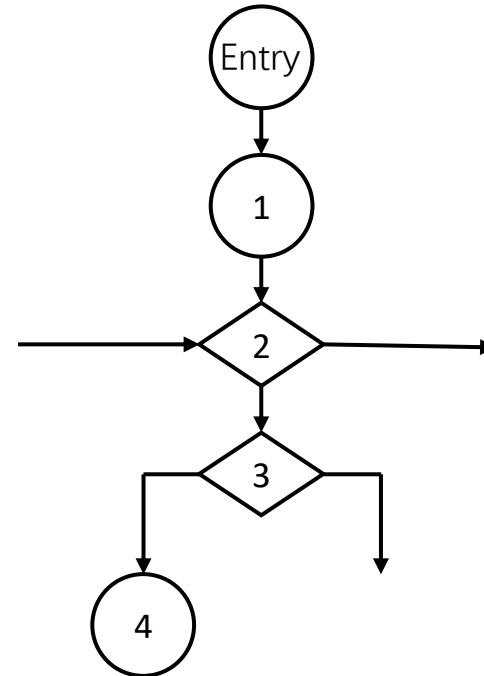
```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11. if( s.charAt(i) == '0')
12.     result = -result;
13. if( result>2147483648 || result<-2147483648)
14.     throw new Exception();
15. return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:
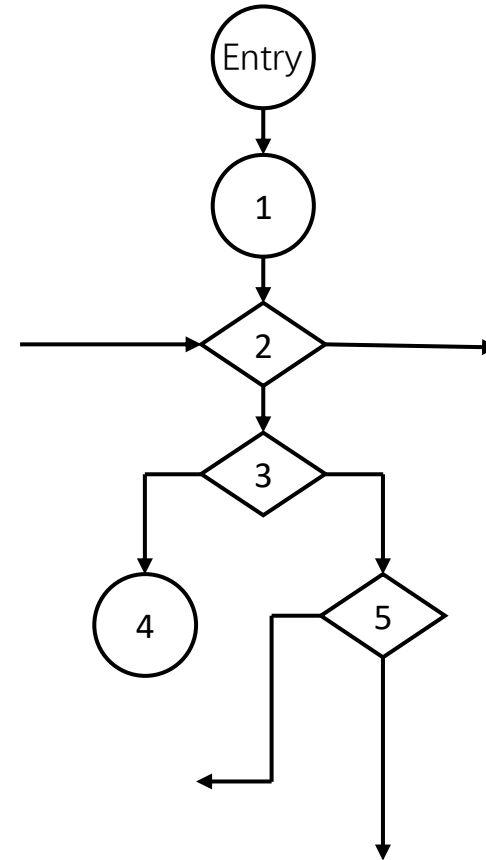
```java
1.     public int Transform(String s) throws Exception{
2.         int result = 0;
3.         int power = 0;
4.         for (int i = s.length() - 1; i >= 1; i--){
5.             if(s.charAt(i) == '1')
6.                 result += Math.pow(2,power);
7.             else if(s.charAt(i) != '0')
8.                     throw new Exception();
9.             power ++;
10.        }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:
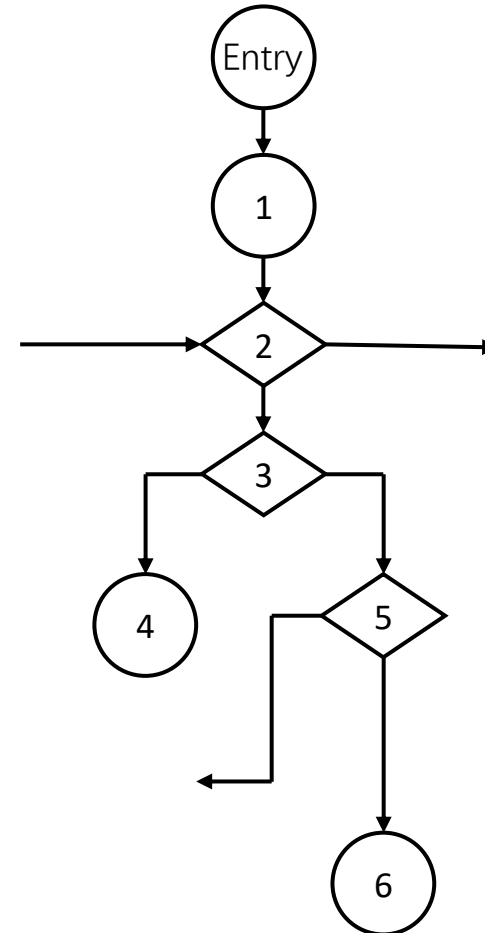
```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                  throw new Exception();
9.              power ++;
10.         }
11.     if( s.charAt(i) == '0')
12.         result = -result;
13.     if( result>2147483648 || result<-2147483648)
14.         throw new Exception();
15.     return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                  throw new Exception();
9.              power ++;
10.         }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:
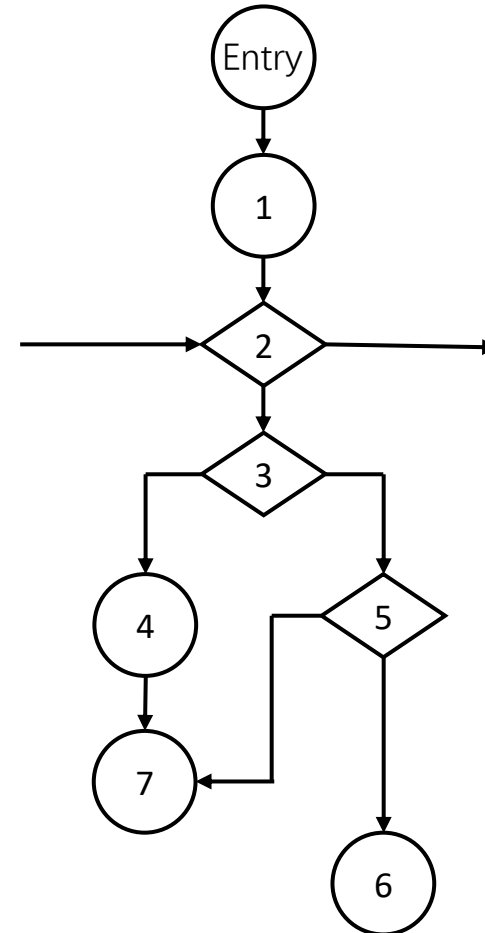
```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.          power ++;
10.     }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:
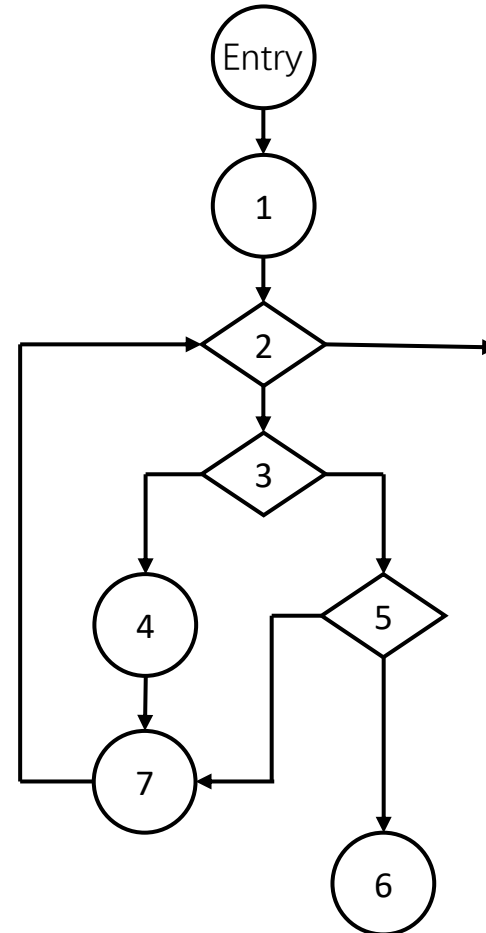
```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.         }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:
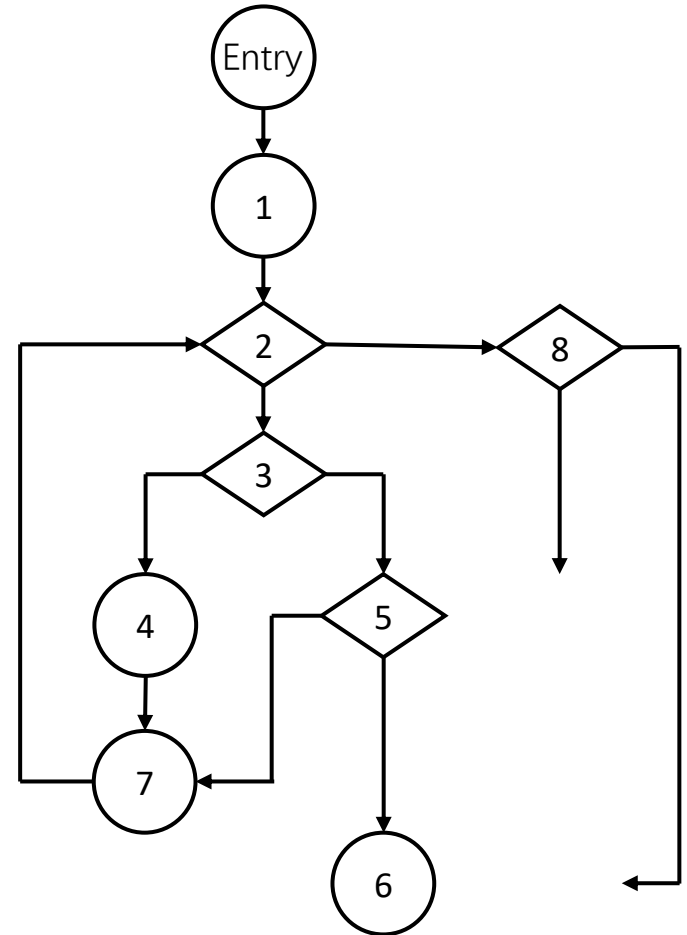
```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11.  if( s.charAt(i) == '0')
12.     result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.     throw new Exception();
15.  return result;
16.}
```

Program Testing\Answer 1

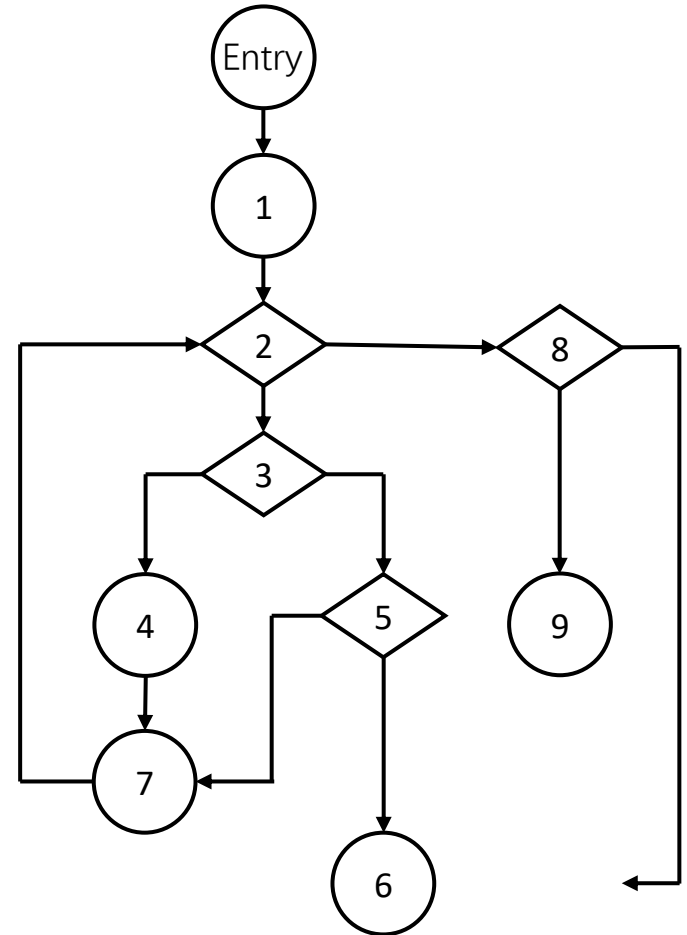Given the following Java program:

```java
1.    public int Transform(String s) throws Exception{
2.        int result = 0;
3.        int power = 0;
4.        for (int i = s.length() - 1; i >= 1; i--){
5.            if(s.charAt(i) == '1')
6.                result += Math.pow(2,power);
7.            else if(s.charAt(i) != '0')
8.                throw new Exception();
9.            power ++;
10.       }
11.  if( s.charAt(i) == '0')
12.       result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.       throw new Exception();
15.  return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:
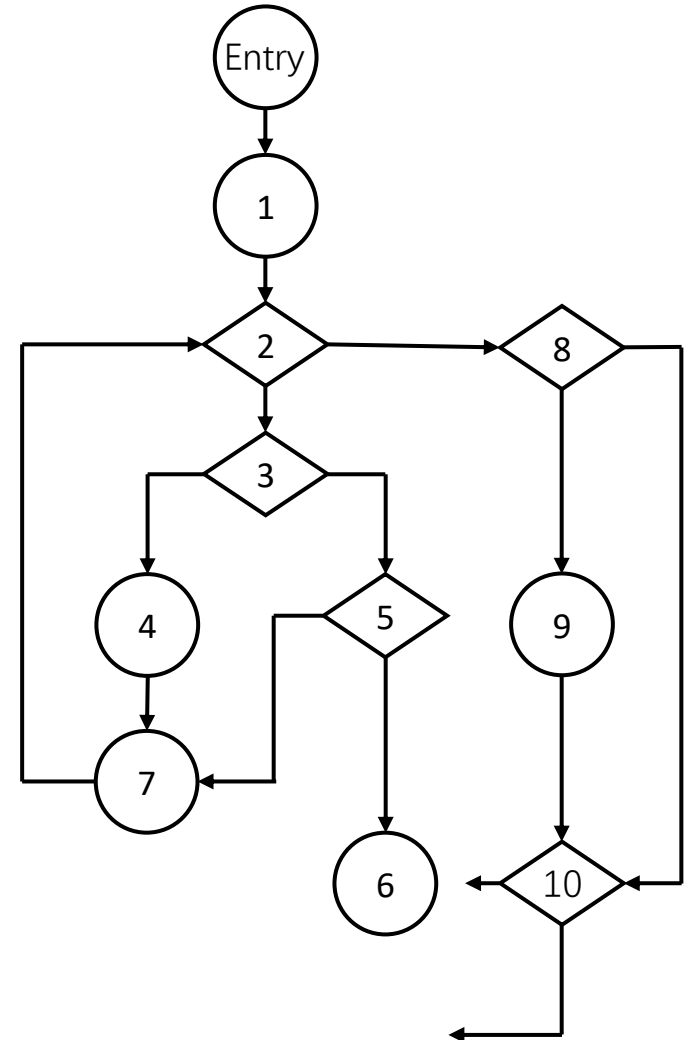
```java
1.     public int Transform(String s) throws Exception{
2.         int result = 0;
3.         int power = 0;
4.         for (int i = s.length() - 1; i >= 1; i--){
5.             if(s.charAt(i) == '1')
6.                 result += Math.pow(2,power);
7.             else if(s.charAt(i) != '0')
8.                 throw new Exception();
9.             power ++;
10.        }
11.    if( s.charAt(i) == '0')
12.        result = -result;
13.    if( result>2147483648 || result<-2147483648)
14.        throw new Exception();
15.    return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:
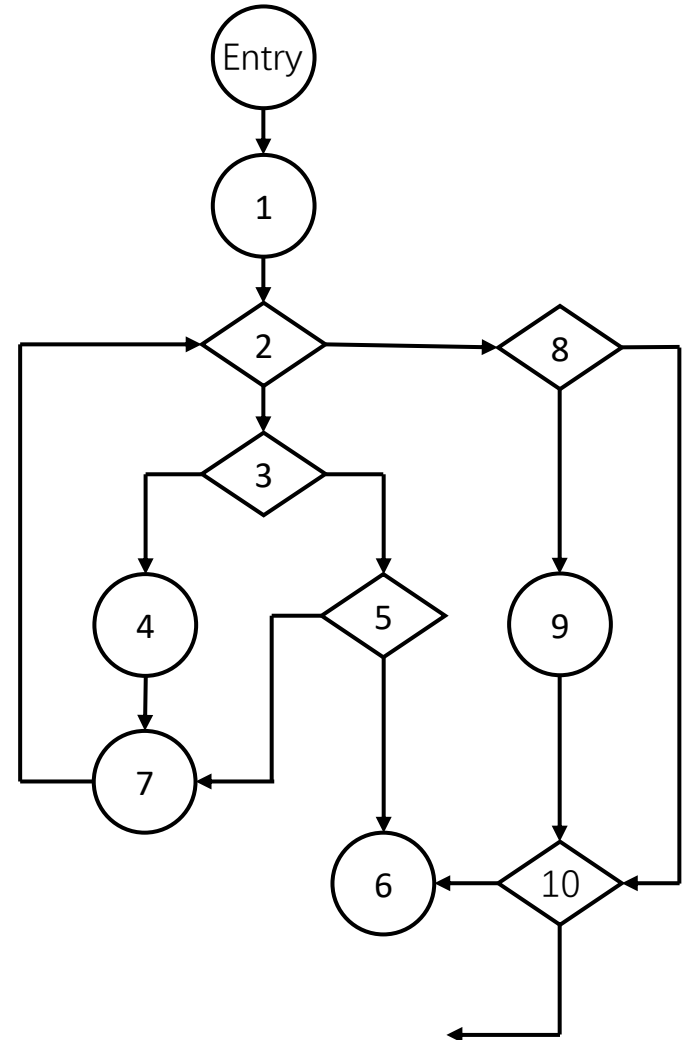
```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                  throw new Exception();
9.              power ++;
10.         }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:
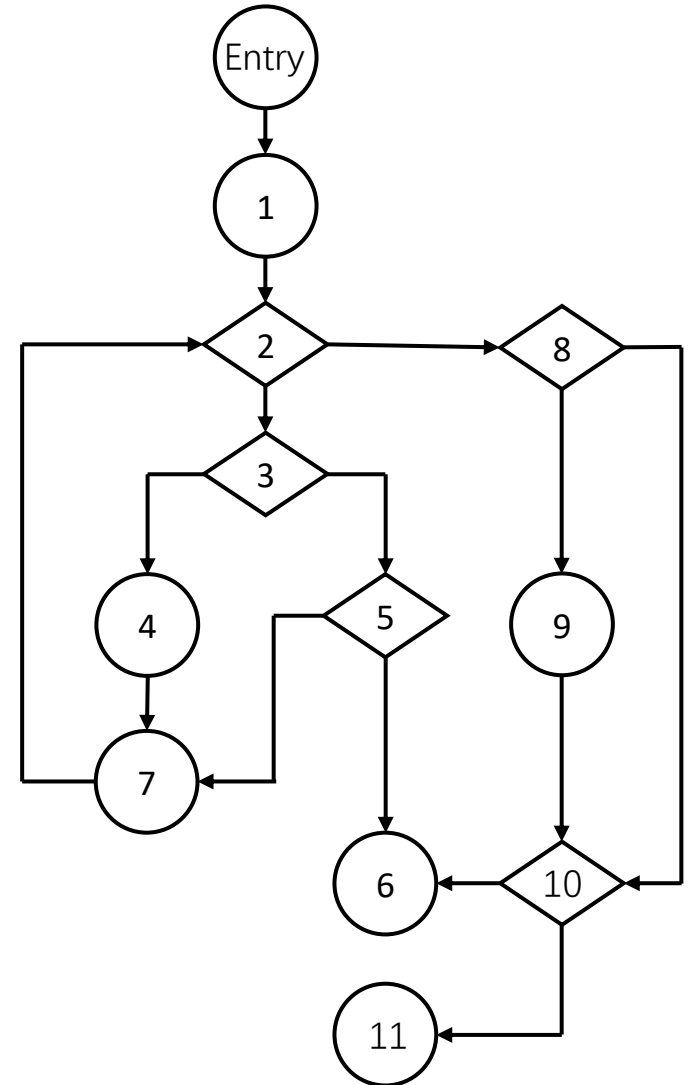
```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                  throw new Exception();
9.              power ++;
10.     }
11. if( s.charAt(i) == '0')
12.     result = -result;
13. if( result>2147483648 || result<-2147483648)
14.     throw new Exception();
15. return result;
16.}
```

Program Testing\Answer 1

Given the following Java program:

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11. if( s.charAt(i) == '0')
12.     result = -result;
13. if( result>2147483648 || result<-2147483648)
14.     throw new Exception();
15. return result;
16.}
```
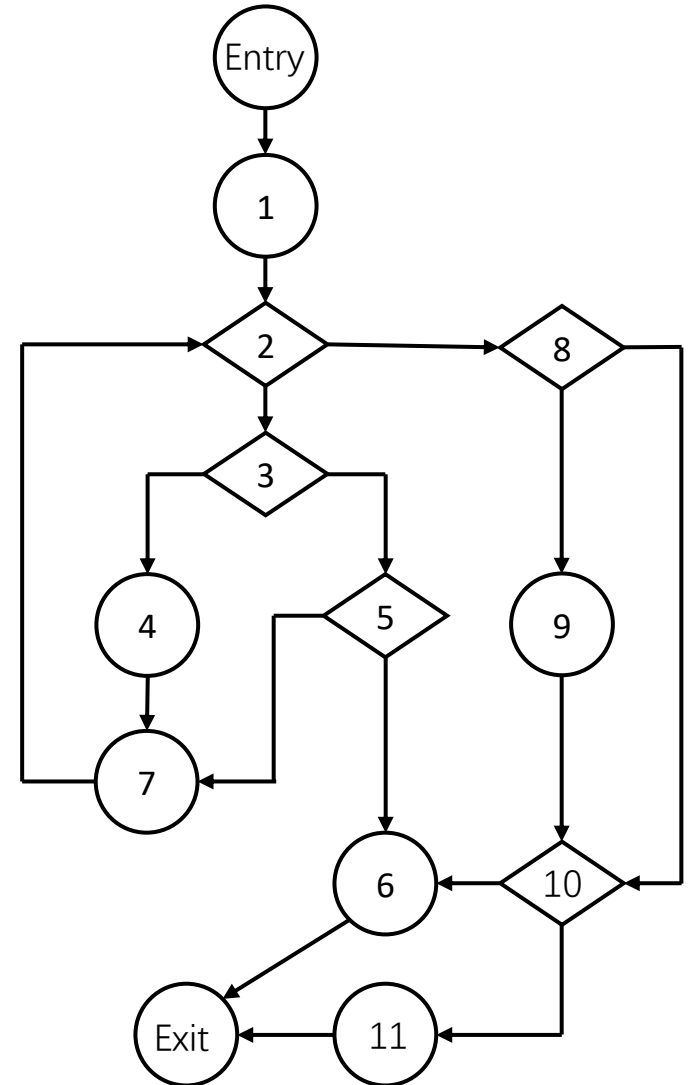

Draw the exit node at last!

Program Testing\Question 2

Calculate the following metrics of the above program according to the control flow graph (CFG) that you draw.
(a)# nodes
(b)# edges
(c)# predicate nodes
(d)# regions
(e)V(G) (McCabe's cyclomatic complexity)

Program Testing\Answer 2
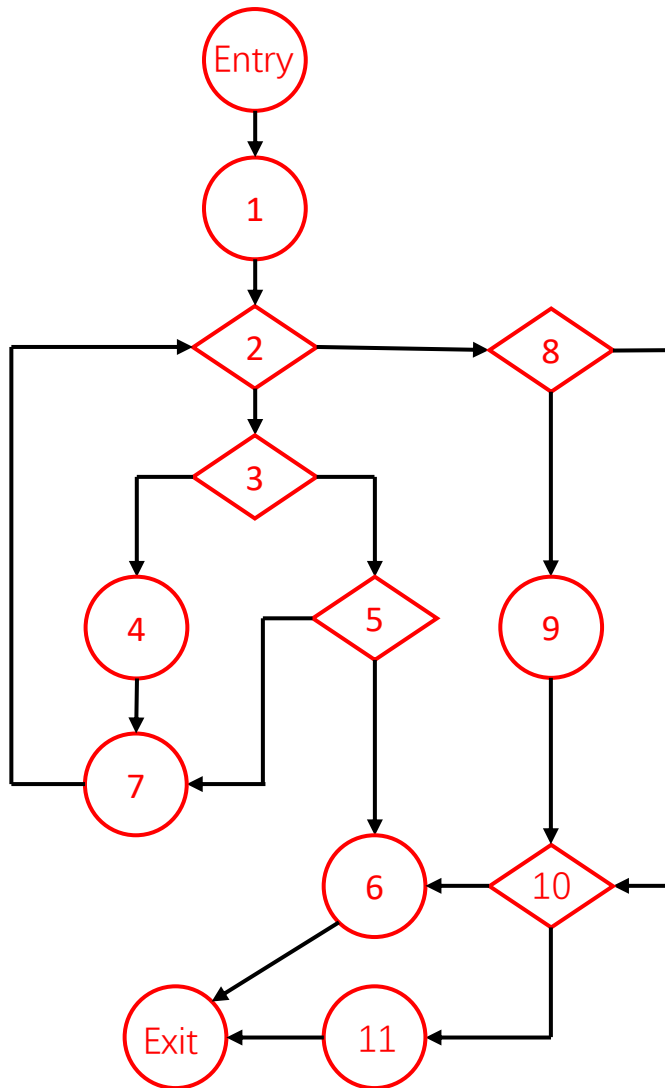
Nodes: 13
Edges:
Predicate nodes:
Regions:
V(G):

Nodes: 13
Edges: 17
Predicate nodes:
Regions:
V(G):

Program Testing\Answer 2
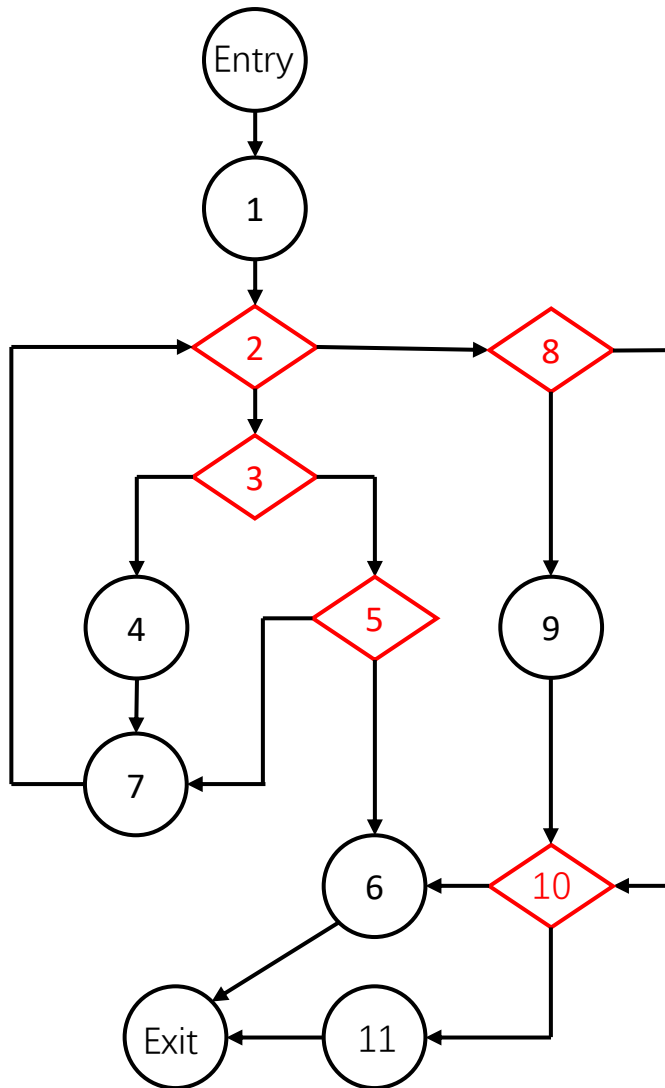
Nodes: 13
Edges: 17
Predicate nodes: 5
Regions:
V(G):

Program Testing\Answer 2
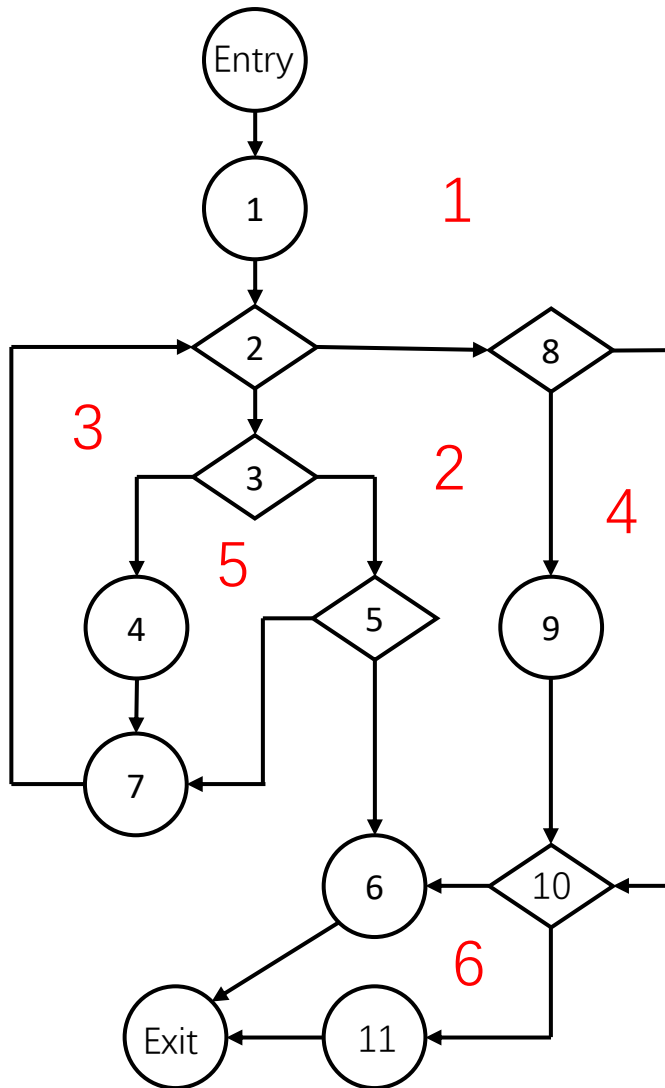
Nodes: 13
Edges: 17
Predicate nodes: 5
Regions: 6
V(G):

Program Testing\Answer 2
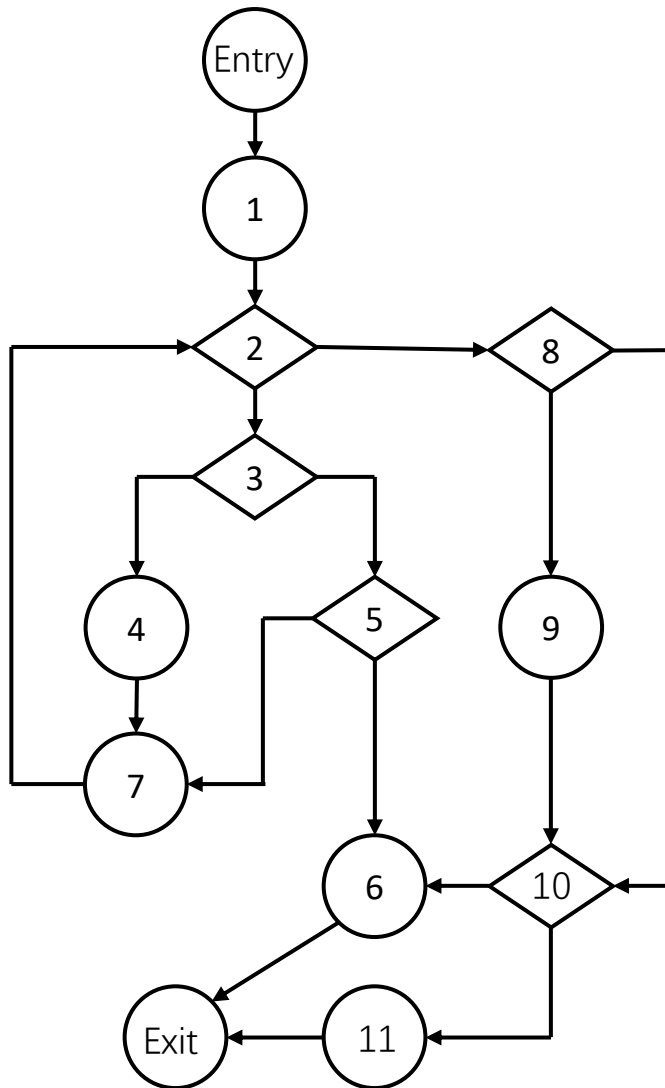
Nodes: 13
Edges: 17
Predicate nodes: 5
Regions: 6
V(G): 6

Program Testing\Answer 2
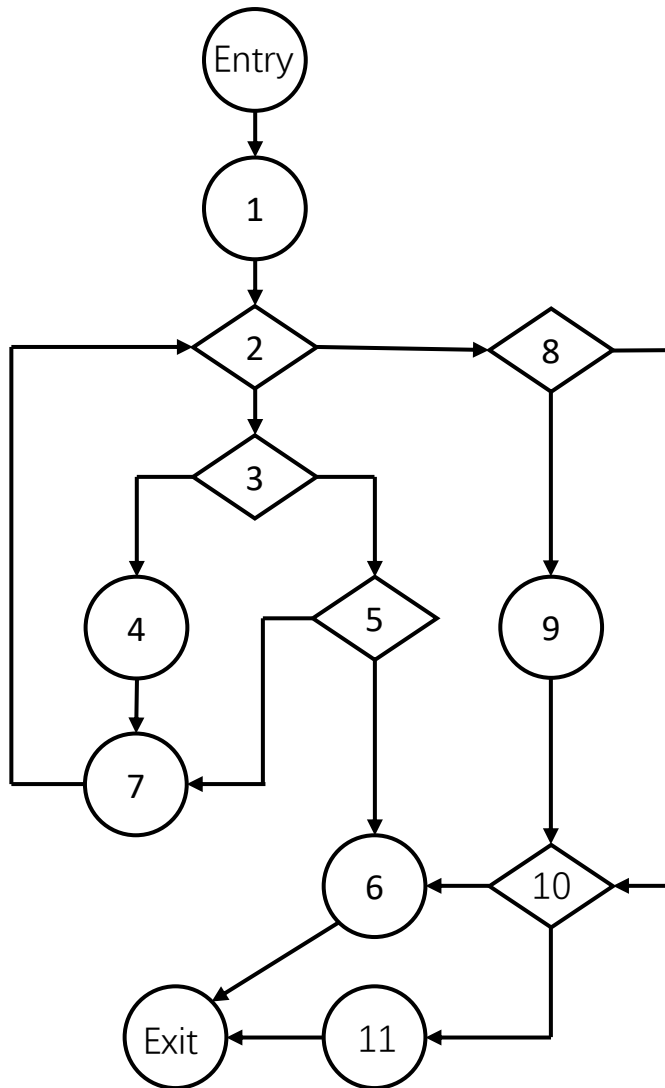
Nodes: 13
Edges: 17
Predicate nodes: 5
Regions: 6
V(G): 6

Program Testing\Question 3

What is the relation among above five metrics?
Also, how many independent paths can be found in the control flow graph that satisfy the edge coverage criterion C1? Please give the complete set of independent paths.

Program Testing\Answer 3

Nodes: 13
Edges: 17
Predicate nodes: 5
Regions: 6
V(G): 6

What is the relation among above five metrics?

Program Testing\Answer 3

Nodes: 13
Edges: 17
Predicate nodes: 5
Regions: 6
V(G): 6

What is the relation among above five metrics?

$$V(G) = E - N + 2 = P + 1$$
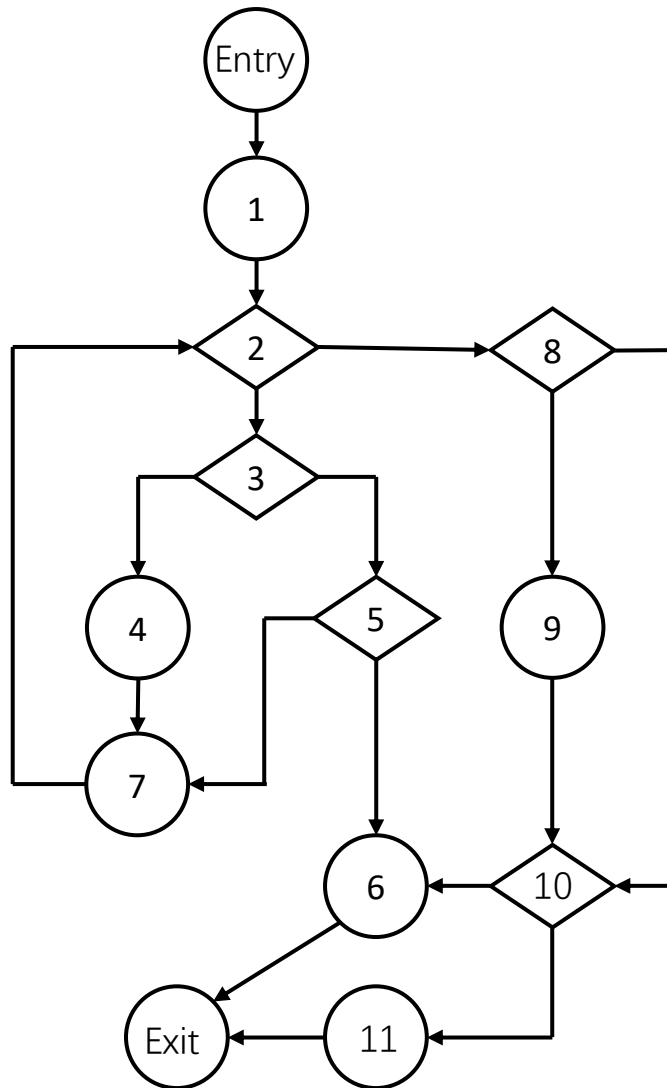
How many independent paths can be found in the control flow graph that satisfy the edge coverage criterion C1?

Program Testing\Answer 3
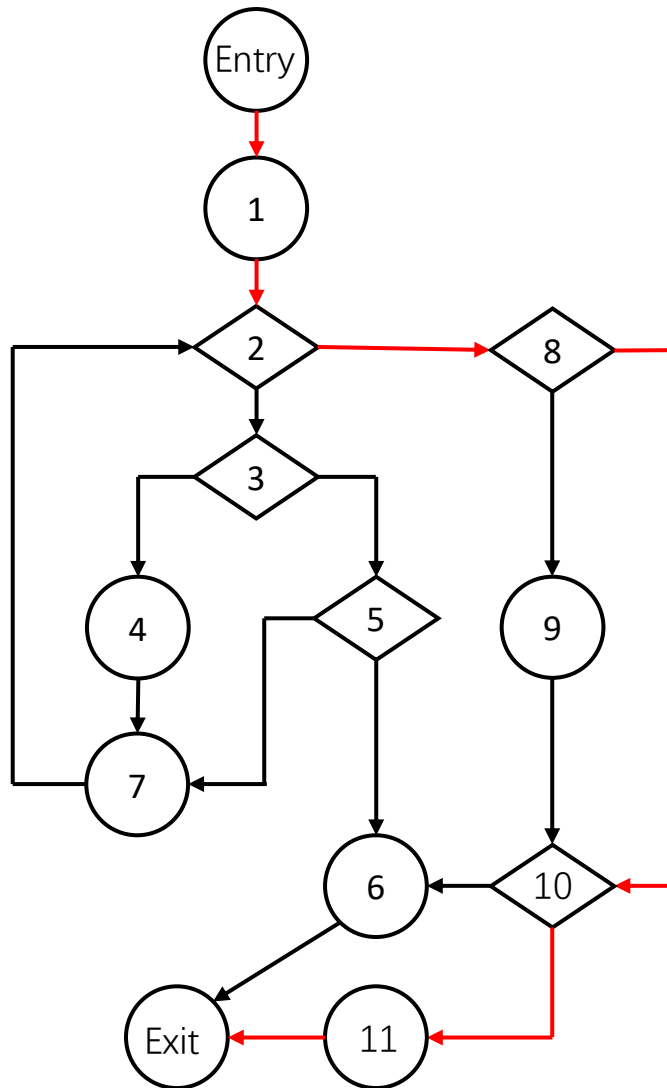
How many independent paths can be found in the control flow graph that satisfy the edge coverage criterion C1?

Entry,1,2,8,10,11,Exit

Program Testing\Answer 3
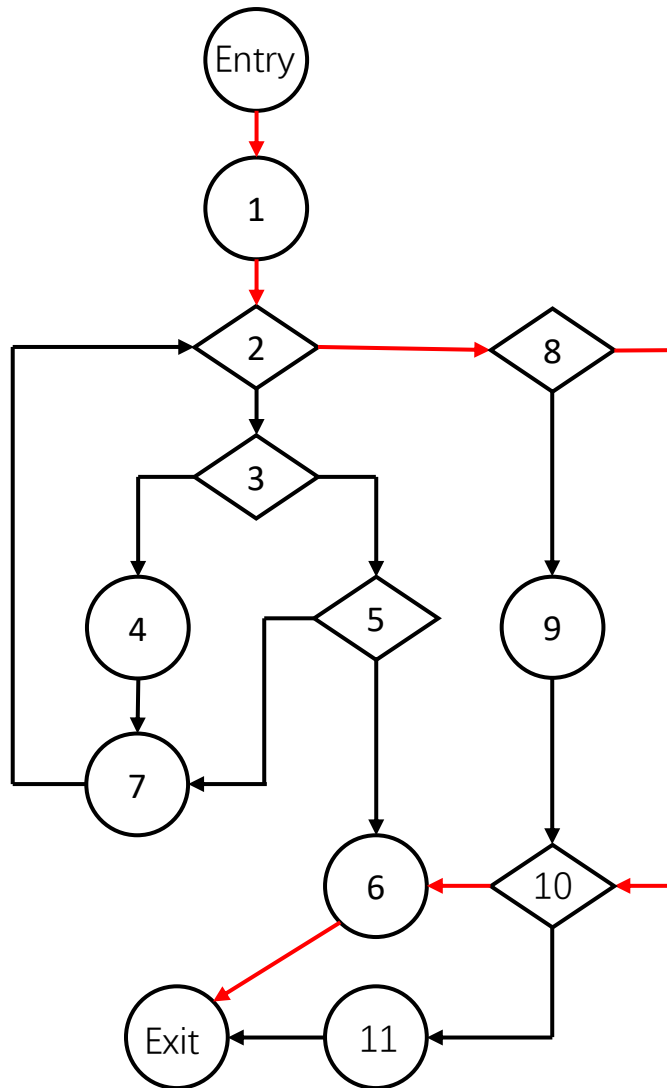
How many independent paths can be found in the control flow graph that satisfy the edge coverage criterion C1?

Entry,1,2,8,10,11,Exit
Entry,1,2,8,10,6,Exit

Program Testing\Answer 3
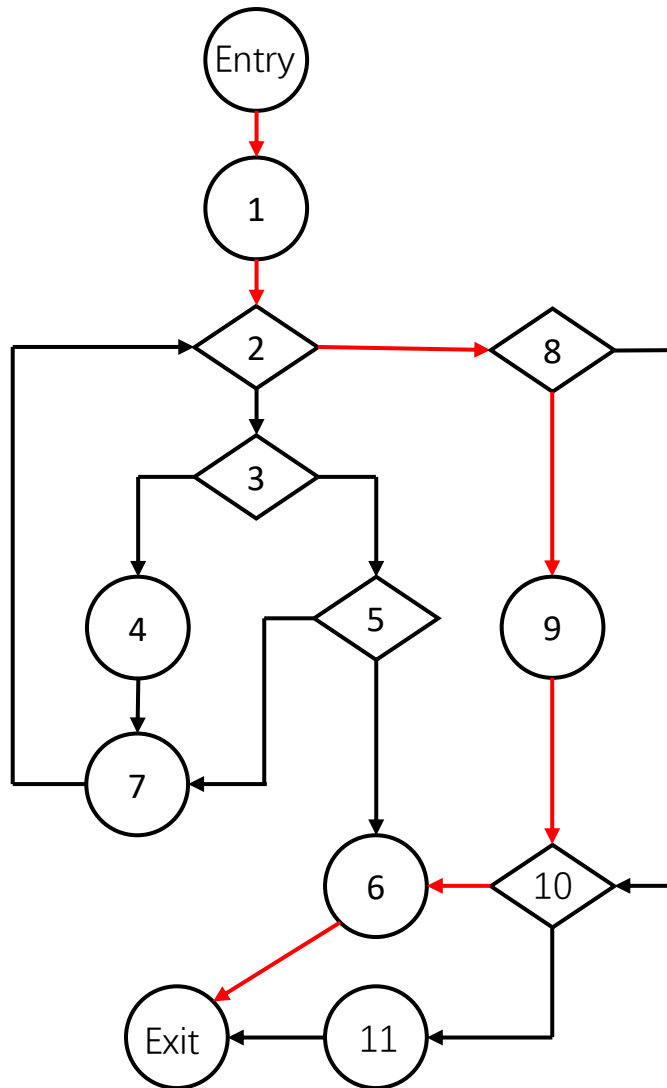
How many independent paths can be found in the control flow graph that satisfy the edge coverage criterion C1?

Entry,1,2,8,10,11,Exit
Entry,1,2,8,10,6,Exit
Entry,1,2,8,9,10,6,Exit

Program Testing\Answer 3
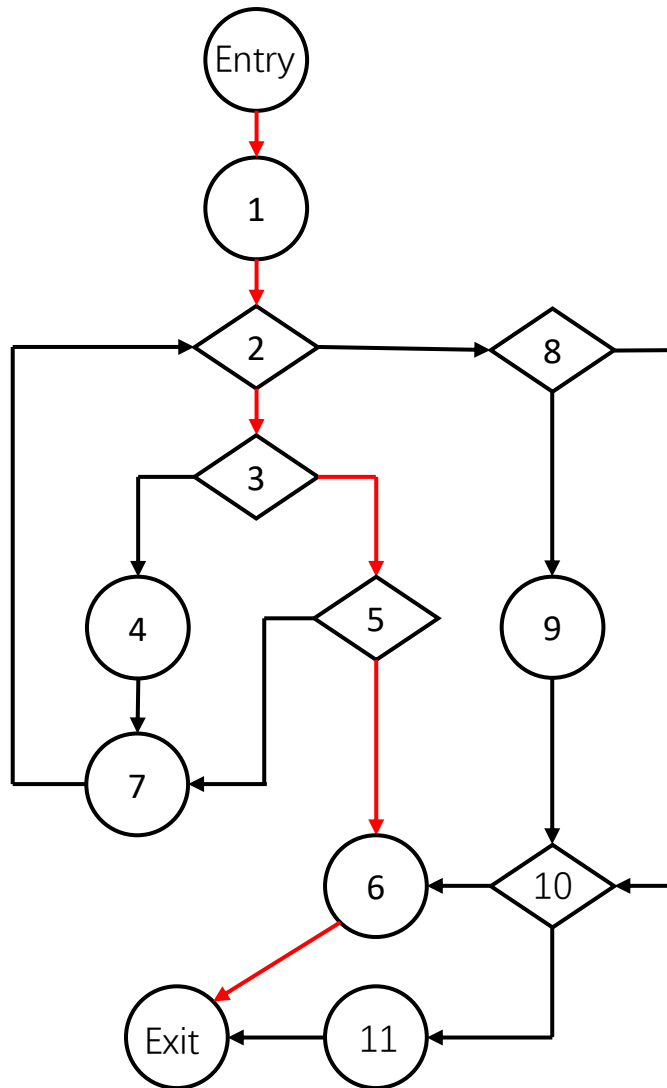
How many independent paths can be found in the control flow graph that satisfy the edge coverage criterion C1?

Entry,1,2,8,10,11,Exit
Entry,1,2,8,10,6,Exit
Entry,1,2,8,9,10,6,Exit
Entry,1,2,3,5,6,Exit

Program Testing\Answer 3
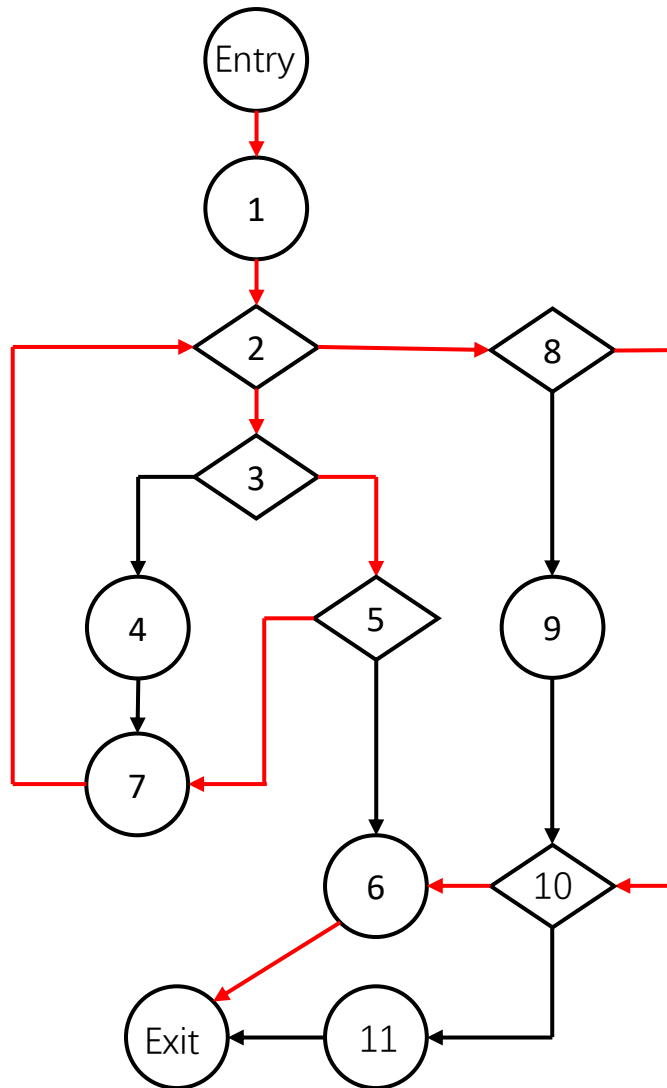
How many independent paths can be found in the control flow graph that satisfy the edge coverage criterion C1?

Entry,1,2,8,10,11,Exit
Entry,1,2,8,10,6,Exit
Entry,1,2,8,9,10,6,Exit
Entry,1,2,3,5,6,Exit
Entry,1,2,3,5,7,2,8,10,6,Exit

Program Testing\Answer 3
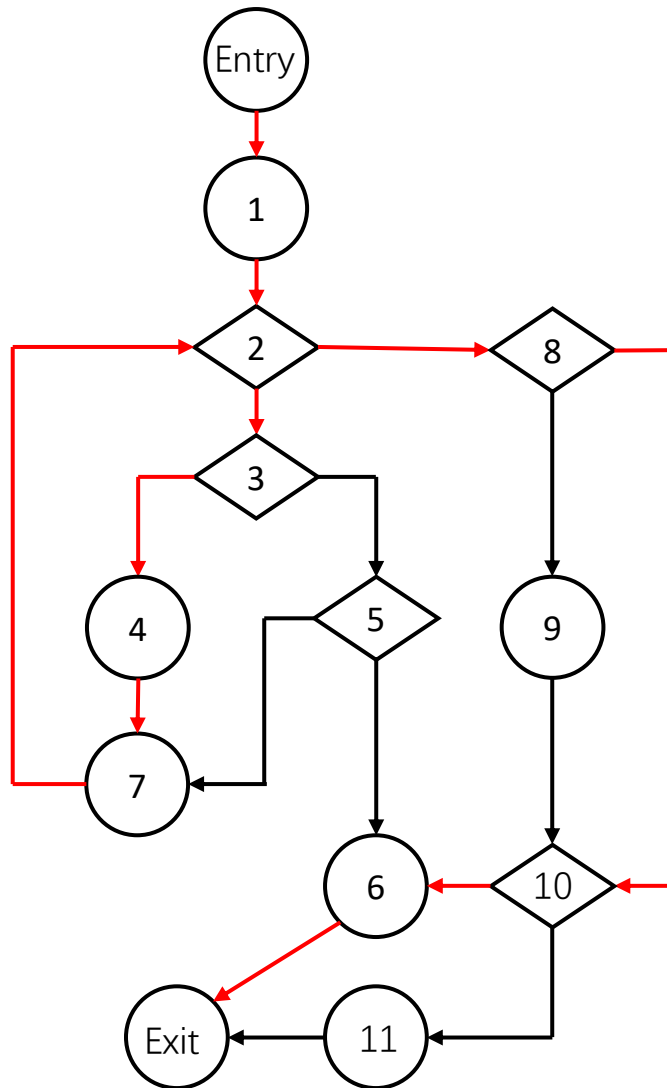
How many independent paths can be found in the control flow graph that satisfy the edge coverage criterion C1?

Entry,1,2,8,10,11,Exit
Entry,1,2,8,10,6,Exit
Entry,1,2,8,9,10,6,Exit
Entry,1,2,3,5,6,Exit
Entry,1,2,3,5,7,2,8,10,6,Exit
Entry,1,2,3,4,7,2,8,10,6,Exit

2. White-box Testing

Source: 2019 / Homework 6 / Question 2

White-box Testing\Question 1

What does the procedure "Transform" do?

```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                  throw new Exception();
9.              power ++;
10.     }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

White-box Testing\Answer 1

What does the procedure "Transform" do?

```java
1.     public int Transform(String s) throws Exception{
2.         int result = 0;
3.         int power = 0;
4.         for (int i = s.length() - 1; i >= 1; i--){
5.             if(s.charAt(i) == '1')
6.                 result += Math.pow(2,power);
7.             else if(s.charAt(i) != '0')
8.                     throw new Exception();
9.             power ++;
10.     }
11. if( s.charAt(i) == '0')
12.     result = -result;
13. if( result>2147483648 || result<-2147483648)
14.     throw new Exception();
15. return result;
16.}
```

The program converts a binary string to integer, where the first element of the string identifies sign of the interger, i.e. 0=negative, 1=positive.

White-box Testing\Question 2

We will use white-box testing to test the program. First, please design a test set to achieve 100% statement coverage.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11.  if( s.charAt(i) == '0')
12.     result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.     throw new Exception();
15.  return result;
16.}
```

White-box Testing\Answer 2

We will use white-box testing to test the program. First, please design a test set to achieve 100% statement coverage.

```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.         }
11.     if( s.charAt(i) == '0')
12.         result = -result;
13.     if( result>2147483648 || result<-2147483648)
14.         throw new Exception();
15.     return result;
16.}
```

White-box Testing\Answer 2

We will use white-box testing to test the program. First, please design a test set to achieve 100% statement coverage.

```java
1.    public int Transform(String s) throws Exception{
2.        int result = 0;
3.        int power = 0;
4.        for (int i = s.length() - 1; i >= 1; i--){
5.            if(s.charAt(i) == '1')
6.                result += Math.pow(2,power);
7.            else if(s.charAt(i) != '0')
8.                    throw new Exception();
9.            power ++;
10.       }
11.    if( s.charAt(i) == '0')
12.        result = -result;
13.    if( result>2147483648 || result<-2147483648)
14.        throw new Exception();
15.    return result;
16.}
```

Test Case:

Test1: 1211

White-box Testing\Answer 2

We will use white-box testing to test the program. First, please design a test set to achieve 100% statement coverage.

```
1.     public int Transform(String s) throws Exception{
2.         int result = 0;
3.         int power = 0;
4.         for (int i = s.length() - 1; i >= 1; i--){
5.             if(s.charAt(i) == '1')
6.                 result += Math.pow(2,power);
7.             else if(s.charAt(i) != '0')
8.                     throw new Exception();
9.             power ++;
10.        }
11.   if( s.charAt(i) == '0')
12.        result = -result;
13.   if( result>2147483648 || result<-2147483648)
14.        throw new Exception();
15.   return result;
16.}
```

Test Case:
   Test1: 1211
   Test2: 01…11 (33 characters long)

White-box Testing\Answer 2

We will use white-box testing to test the program. First, please design a test set to achieve 100% statement coverage.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11. if( s.charAt(i) == '0')
12.     result = -result;
13. if( result>2147483648 || result<-2147483648)
14.     throw new Exception();
15. return result;
16.}
```

Test Case:
   Test1: 1211
   Test2: 01…11 (33 characters long)
   Test3: 1110 (line 15)

White-box Testing\Question 3

Can we achieve 100% edge coverage? If so, please design additional test sets, otherwise, give the reasons.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11. if( s.charAt(i) == '0')
12.     result = -result;
13. if( result>2147483648 || result<-2147483648)
14.     throw new Exception();
15. return result;
16.}
```

White-box Testing\Answer 3

Can we achieve 100% edge coverage? If so, please design additional test sets, otherwise, give the reasons.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){          ➡ False, True
5.              if(s.charAt(i) == '1')                           ➡ False, True
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')                      ➡ False, True
8.                      throw new Exception();
9.              power ++;
10.         }
11.     if( s.charAt(i) == '0')                                  ➡ False, True
12.         result = -result;
13.     if( result>2147483648 || result<-2147483648)            ➡ False, True
14.         throw new Exception();
15.  return result;
16.}
```

White-box Testing\Answer 3

Can we achieve 100% edge coverage? If so, please design additional test sets, otherwise, give the reasons.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){         ➡  False, True
5.              if(s.charAt(i) == '1')                          ➡  False, True
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')                     ➡  False, True
8.                      throw new Exception();
9.              power ++;
10.         }
11.     if( s.charAt(i) == '0')                                 ➡  False, True
12.         result = -result;
13.     if( result>2147483648 || result<-2147483648)           ➡  False, True
14.         throw new Exception();
15.   return result;
16.}
```

Test Case:
Test1: 1211
Test2: 01…11 (33 characters long)
Test3: 1110 (line 15)

White-box Testing\Answer 3

Can we achieve 100% edge coverage? If so, please design additional test sets, otherwise, give the reasons.

```
1.       public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){        ➡ False, True
5.              if(s.charAt(i) == '1')                          ➡ False, True
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')                     ➡ False, True
8.                      throw new Exception();
9.              power ++;
10.         }
11.     if( s.charAt(i) == '0')                                 ➡ False, True
12.         result = -result;
13.     if( result>2147483648 || result<-2147483648)           ➡ False, True
14.         throw new Exception();
15.   return result;
16.}
```

Test Case:

Test1: 1211

Test2: 01…11 (33 characters long)

Test3: 1110 (line 15)

White-box Testing\Answer 3

Can we achieve 100% edge coverage? If so, please design additional test sets, otherwise, give the reasons.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){      ➡ False, True
5.              if(s.charAt(i) == '1')                        ➡ False, True
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')                   ➡ False, True
8.                  throw new Exception();
9.              power ++;
10.         }
11.     if( s.charAt(i) == '0')                               ➡ False, True
12.         result = -result;
13.     if( result>2147483648 || result<-2147483648)         ➡ False, True
14.         throw new Exception();
15.     return result;
16.}
```

Test Case:
Test1: 1211
Test2: 01…11 (33 characters long)
Test3: 1110 (line 15)

White-box Testing\Answer 3

Can we achieve 100% edge coverage? If so, please design additional test sets, otherwise, give the reasons.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){        ➡ False, True
5.              if(s.charAt(i) == '1')                          ➡ False, True
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')                     ➡ False, True
8.                      throw new Exception();
9.              power ++;
10.         }
11.     if( s.charAt(i) == '0')                                 ➡ False, True
12.         result = -result;
13.     if( result>2147483648 || result<-2147483648)           ➡ False, True
14.         throw new Exception();
15.     return result;
16.}
```

All edges are included!

White-box Testing\Question 4

Also, can we achieve 100% condition coverage and path coverage respectively? If so, please design the additional test sets, otherwise, give the reasons. (You only need to specify the input values of each test.) You should first answer Yes/No, and then give the test sets/reasons.

```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.         }
11.   if( s.charAt(i) == '0')
12.       result = -result;
13.   if( result>2147483648 || result<-2147483648)
14.       throw new Exception();
15.   return result;
16.}
```

White-box Testing\Answer 4

Also, can we achieve 100% condition coverage and path coverage respectively? If so, please design the additional test sets, otherwise, give the reasons. (You only need to specify the input values of each test.) You should first answer Yes/No, and then give the test sets/reasons.

```java
1.     public int Transform(String s) throws Exception{
2.         int result = 0;
3.         int power = 0;
4.         for (int i = s.length() - 1; i >= 1; i--){
5.             if(s.charAt(i) == '1')
6.                 result += Math.pow(2,power);
7.             else if(s.charAt(i) != '0')
8.                     throw new Exception();
9.             power ++;
10.        }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Only one conditional statement contains two conditions!

White-box Testing\Answer 4

Also, can we achieve 100% condition coverage and path coverage respectively? If so, please design the additional test sets, otherwise, give the reasons. (You only need to specify the input values of each test.) You should first answer Yes/No, and then give the test sets/reasons.

```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                  throw new Exception();
9.              power ++;
10.         }
11.     if( s.charAt(i) == '0')
12.         result = -result;
13.     if( result>2147483648 || result<-2147483648)
14.         throw new Exception();
15.     return result;
16.}
```

Test Case:
Test1: 1211
Test2: 01…11 (33 characters long)
Test3: 1110 (line 15)

Line 13 → (False, True) || (False, True)

Also, can we achieve 100% condition coverage and path coverage respectively? If so, please design the additional test sets, otherwise, give the reasons. (You only need to specify the input values of each test.) You should first answer Yes/No, and then give the test sets/reasons.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.         }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Test Case:

Test1: 1211

Test2: 01…11 (33 characters long)

Test3: 1110 (line 15)

Line 13: (False, True) || (False, True)

White-box Testing\Answer 4

Also, can we achieve 100% condition coverage and path coverage respectively? If so, please design the additional test sets, otherwise, give the reasons. (You only need to specify the input values of each test.) You should first answer Yes/No, and then give the test sets/reasons.

```java
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                  throw new Exception();
9.              power ++;
10.         }
11.     if( s.charAt(i) == '0')
12.         result = -result;
13.     if( result>2147483648 || result<-2147483648)
14.         throw new Exception();
15.     return result;
16.}
```

Test Case:

Test1: 1211

Test2: 01…11 (33 characters long)

Test3: 1110 (line 15)

Line 13 → (False, True) || (False, True)

White-box Testing\Answer 4

Also, can we achieve 100% condition coverage and path coverage respectively? If so, please design the additional test sets, otherwise, give the reasons. (You only need to specify the input values of each test.) You should first answer Yes/No, and then give the test sets/reasons.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                  throw new Exception();
9.              power ++;
10.         }
11.  if( s.charAt(i) == '0')
12.      result = -result;
13.  if( result>2147483648 || result<-2147483648)
14.      throw new Exception();
15.  return result;
16.}
```

Result > 2147483648 should be covered!

Additional Test: 11…111 (result > 2147483648)

(False, True) || (False, True)

White-box Testing\Answer 4

Also, can we achieve 100% condition coverage and path coverage respectively? If so, please design the additional test sets, otherwise, give the reasons. (You only need to specify the input values of each test.) You should first answer Yes/No, and then give the test sets/reasons.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.         }
11.     if( s.charAt(i) == '0')
12.         result = -result;
13.     if( result>2147483648 || result<-2147483648)
14.         throw new Exception();
15.     return result;
16.}
```

How about path coverage?

No! Since s.length() is unbounded!

3. Black-box Testing

Source: 2019 / Homework 6 / Question 3

Black-box Testing\Question 1

Referring to **Problem 1**

Please design a test set to perform black-box testing. State clearly your testing strategy and corresponding test inputs (Note: The test set should include at least 4 test cases).

Black-box Testing\Answer 1

Please design a test set to perform black-box testing. State clearly your testing strategy and corresponding test inputs (Note: The test set should include at least 4 test cases).

Testing strategy

(1) Generate an empty string.

(2) Generate a binary string of short length.

(3) Generate a binary string of long length.

(4) Generate a binary string with the first element '0'

(5) Generate a binary string with the first element '1'

(6) Generate a string with invalid characters except '0' and '1'

Black-box Testing\Question 2

There are some bugs in the program that may or may not throw an exception. Please point out those bugs and design proper input strings that may incur the exception.

```
1.      public int Transform(String s) throws Exception{
2.          int result = 0;
3.          int power = 0;
4.          for (int i = s.length() - 1; i >= 1; i--){
5.              if(s.charAt(i) == '1')
6.                  result += Math.pow(2,power);
7.              else if(s.charAt(i) != '0')
8.                      throw new Exception();
9.              power ++;
10.     }
11. if( s.charAt(i) == '0')
12.     result = -result;
13. if( result>2147483648 || result<-2147483648)
14.     throw new Exception();
15. return result;
16.}
```

Black-box Testing\Answer 2

There are some bugs in the program that may or may not throw an exception. Please point out those bugs and design proper input strings that may incur the exception.

```
1.      public int Transform(String s) throws Exception{
2.         int result = 0;
3.         int power = 0;
4.         for (int i = s.length() - 1; i >= 1; i--){
5.             if(s.charAt(i) == '1')
6.                 result += Math.pow(2,power);
7.             else if(s.charAt(i) != '0')
8.                     throw new Exception();
9.             power ++;
10.        }
11.    if( s.charAt(i) == '0')
12.        result = -result;
13.    if( result>2147483648 || result<-2147483648)
14.        throw new Exception();
15.    return result;
16.}
```

result is of int type, and it can be overflow (>2147483648)

Black-box Testing\Answer 2

There are some bugs in the program that may or may not throw an exception. Please point out those bugs and design proper input strings that may incur the exception.

```
1.    public int Transform(String s) throws Exception{
2.        int result = 0;
3.        int power = 0;
4.        for (int i = s.length() - 1; i >= 1; i--){
5.            if(s.charAt(i) == '1')
6.                result += Math.pow(2,power);
7.            else if(s.charAt(i) != '0')
8.                    throw new Exception();
9.            power ++;
10.       }
11.   if( s.charAt(i) == '0')
12.       result = -result;
13.   if( result>2147483648 || result<-2147483648)
14.       throw new Exception();
15.   return result;
16.}
```

result is of int type, and it can be overflow (>2147483648)

i is not defined in line 11

Black-box Testing\Answer 2

There are some bugs in the program that may or may not throw an exception. Please point out those bugs and design proper input strings that may incur the exception.

```
1.     public int Transform(String s) throws Exception{
2.         int result = 0;
3.         int power = 0;
4.         for (int i = s.length() - 1; i >= 1; i--){
5.             if(s.charAt(i) == '1')
6.                 result += Math.pow(2,power);
7.             else if(s.charAt(i) != '0')
8.                 throw new Exception();
9.             power ++;
10.        }
11.    if( s.charAt(i) == '0')
12.        result = -result;
13.    if( result>2147483648 || result<-2147483648)
14.        throw new Exception();
15.    return result;
16.}
```

result is of int type, and it can be overflow (>2147483648)

when i loops to 1, the for loop exits. If the first character is not 1 or 0, for example, input is "#10", it should not return a value.

i is not defined in line 11

Black-box Testing\Question 3

Can black-box testing or white-box testing find the bug? Give the reasons.

Black-box Testing\Answer 3

Can black-box testing or white-box testing find the bug? Give the reasons.

White-box No, Black-box Yes
white-box testing can only guarantee the testing of all program paths. It may find bugs only if all test sets are designed accordingly.
black-box testing can have infinity possibilities, and some corner cases could incur the bugs if well designed.

Black-box Testing\Question 4

Can you compare the white box testing with the black box testing? Please list their characteristics with 2~3 items for each. Can white-box testing find the missing parts (functionalities, etc.) of requirements specification?

Black-box Testing\Answer 4

Can you compare the white box testing with the black box testing? Please list their characteristics with 2~3 items for each. Can white-box testing find the missing parts (functionalities, etc.) of requirements specification?

Criteria	Black Box Testing	White Box Testing
Definition	Black Box Testing is a software testing method in which the internal structure/ design/ implementation of the item being tested is NOT known to the tester	White Box Testing is a software testing method in which the internal structure/ design/ implementation of the item being tested is known to the tester.
Levels Applicable To	Mainly applicable to higher levels of testing: Acceptance Testing System Testing	Mainly applicable to lower levels of testing: Unit Testing Integration Testing
Responsibility	Generally, independent Software Testers	Generally, Software Developers
Programming Knowledge	Not Required	Required
Implementation Knowledge	Not Required	Required
Basis for Test Cases	Requirement Specifications	Detail Design

Black-box Testing\Answer 4

Can you compare the white box testing with the black box testing? Please list their characteristics with 2~3 items for each. Can white-box testing find the missing parts (functionalities, etc.) of requirements specification?

Criteria	Black Box Testing	White Box Testing
Definition	Black Box Testing is a software testing method in which the internal structure/ design/ implementation of the item being tested is NOT known to the tester	White Box Testing is a software testing method in which the internal structure/ design/ implementation of the item being tested is known to the tester.
Levels Applicable To	Mainly applicable to higher levels of testing:Acceptance Testing System Testing	Mainly applicable to lower levels of testing:Unit Testing Integration Testing
Responsibility	Generally, independent Software Testers	Generally, Software Developers
Programming Knowledge	Not Required	Required
Implementation Knowledge	Not Required	Required
Basis for Test Cases	Requirement Specifications	Detail Design

White-box cannot find the missing parts of requirements specification.

4. Code Coverage

Source: 2019 / Homework 6 / Question 4

Code Coverage\Question 1

Given the following program control flow, "Y" and "N" denote Yes and No in the branch, respectively. The lower case characters a-l represent the specific edges between two statements. The number represents the corresponding if-else condition.
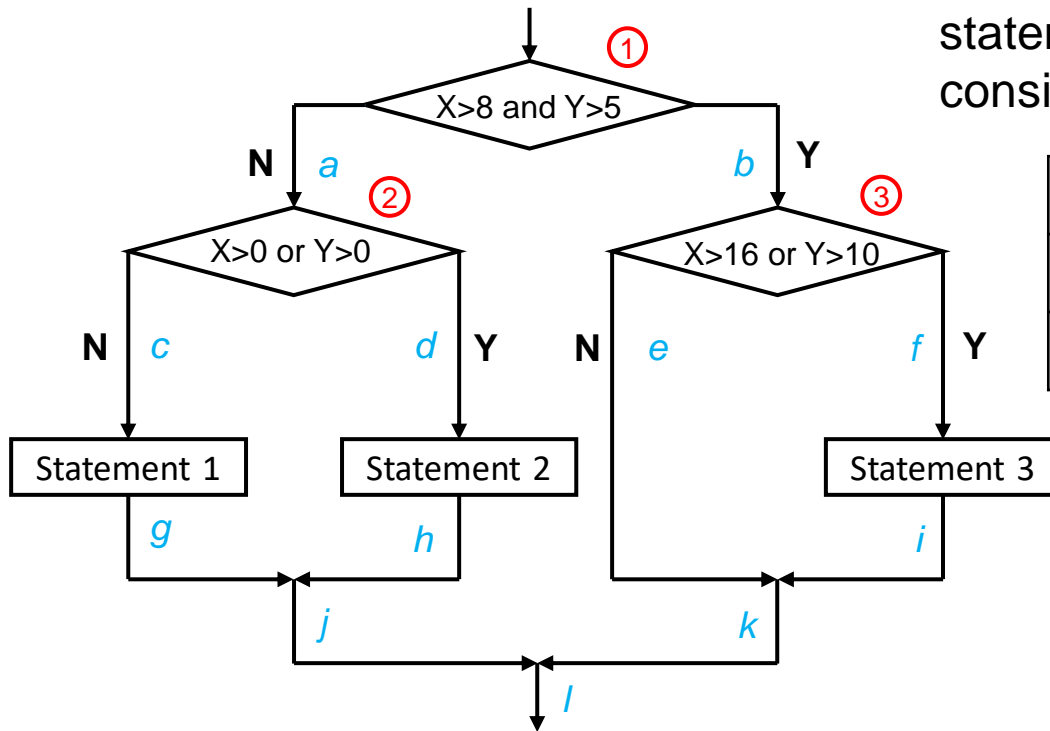
Please design test sets to achieve 100% statement coverage. (Assume that we only consider statements 1,2,3 here)

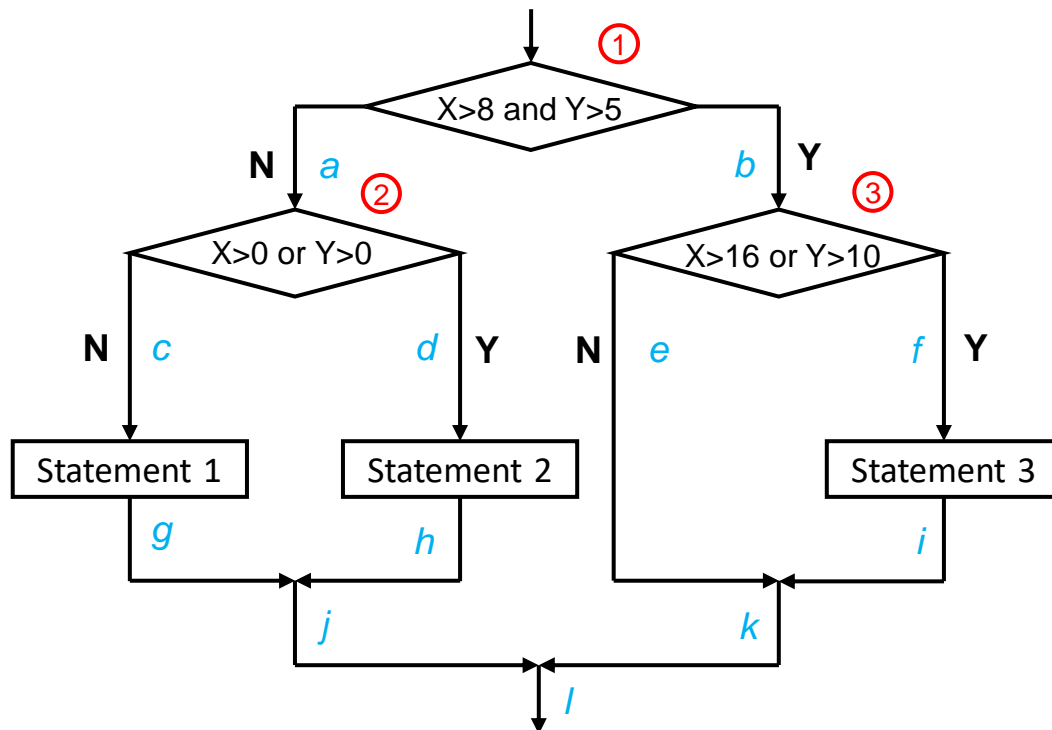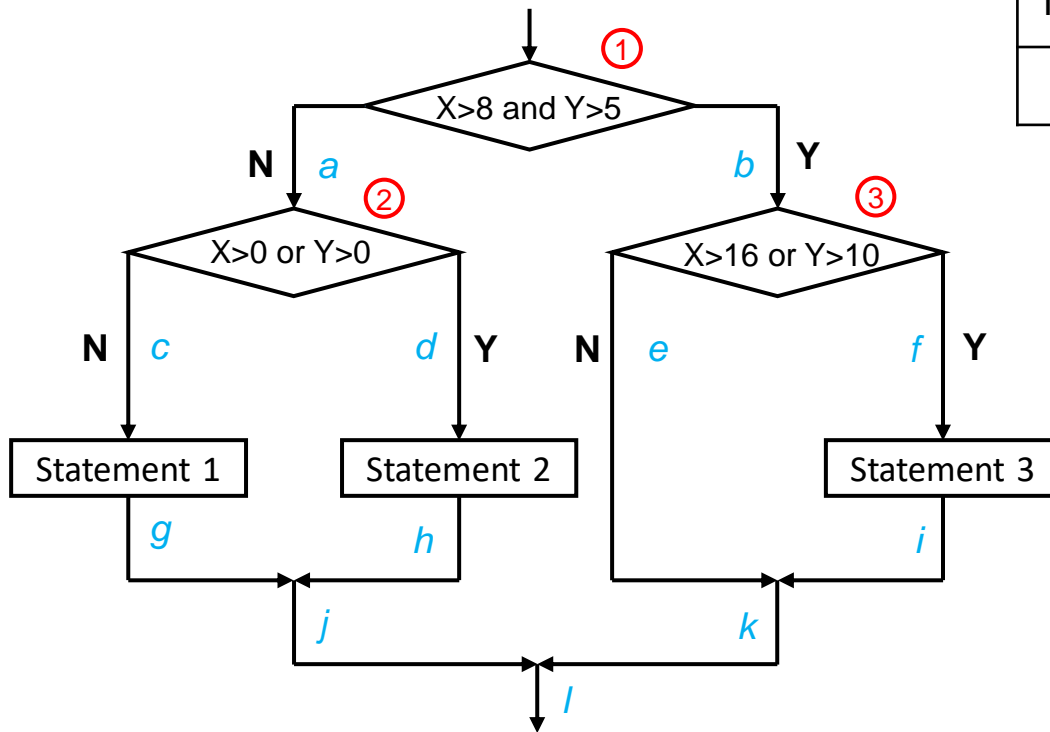| Test Cases	Covered Statements
X=0, Y=0	Statement 1

Code Coverage\Answer 1

Please design test sets to achieve 100% statement coverage. (Assume that we only consider statements 1,2,3 here)

Test Cases	Covered Statements
X=0, Y=0	Statement 1
X=1, Y=0	Statement 2
X=17, Y=6	Statement 3

Code Coverage\Question 2

Please design test sets to achieve 100% branch coverage.

Test Cases	Covered Branches

Code Coverage\Answer 2

Please design test sets to achieve 100% branch coverage.

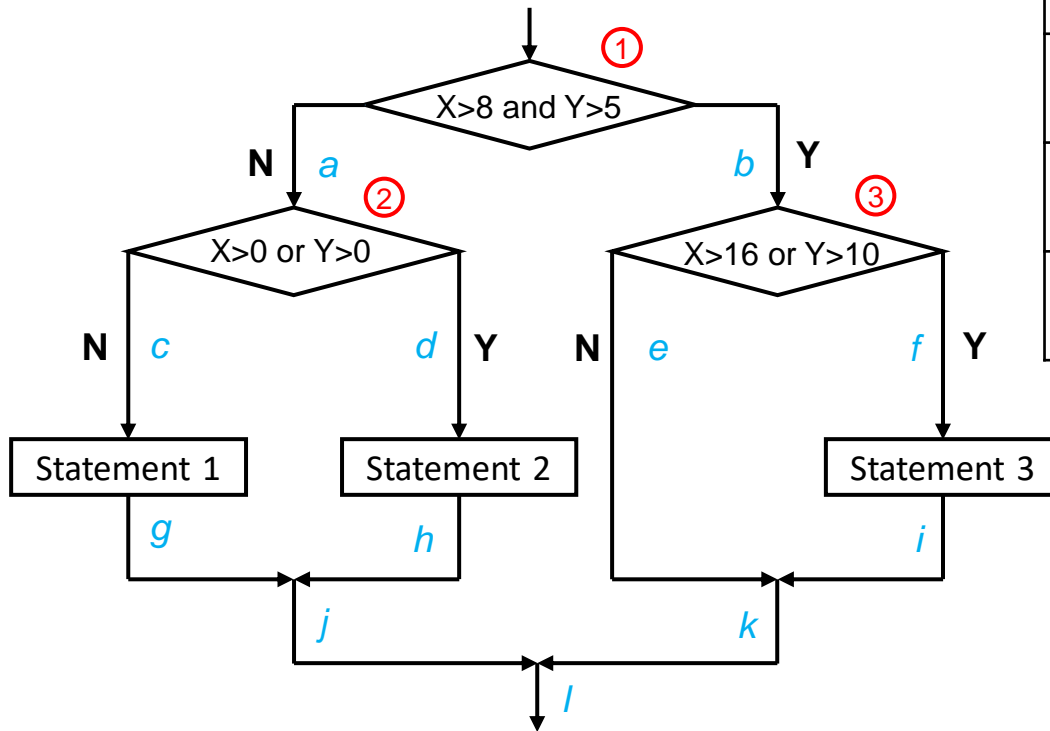| Test Cases	Covered Branches
X=0, Y=0	Condition 1 - N, Condition 2 - N
X=1, Y=0	Condition 1 - N, Condition 2 – Y
X=15, Y=9	Condition 1 - Y, Condition 3 - N
X=17, Y=6	Condition 1 - Y, Condition 3 - Y

Code Coverage\Question 3

Please design test sets to achieve 100% path coverage.

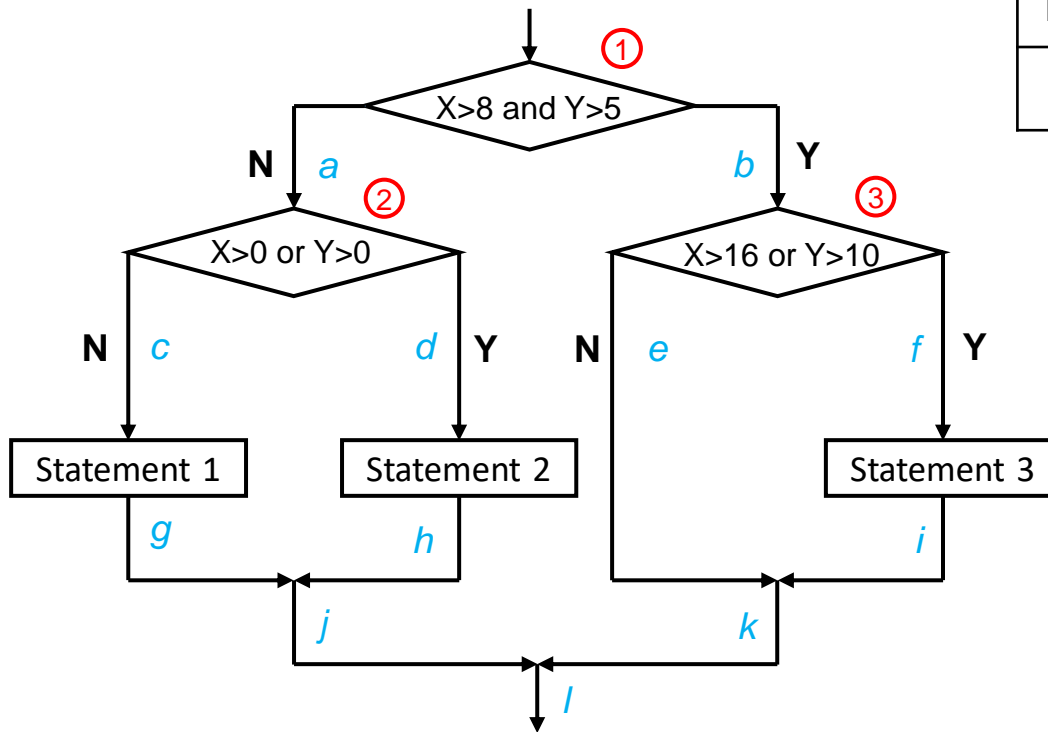| Test Cases	Covered Paths

Code Coverage\Answer 3

Please design test sets to achieve 100% path coverage.
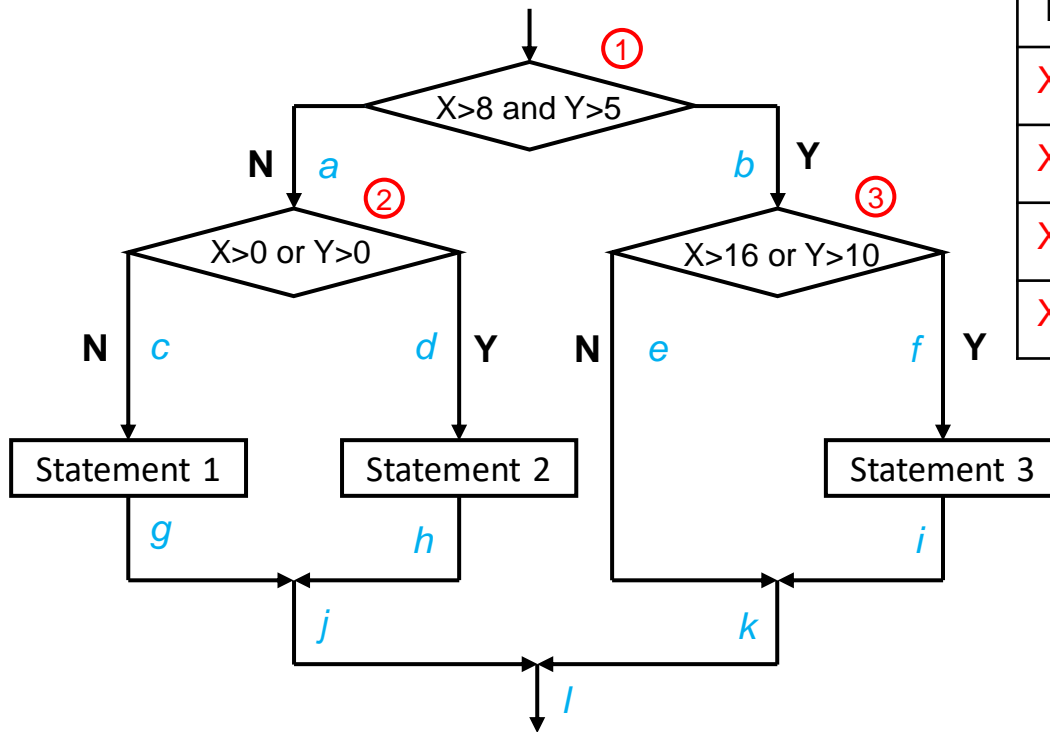
Test Cases	Covered Paths
X=0, Y=0	a→c→g→j→l
X=1, Y=1	a→d→h→j→l
X=15, Y=9	b→e→k→l
X=17, Y=11	b→f→i→k→l

5. Stub and Driver Modules

Source: 2020 / Homework 6 / Question 4

Stub and Driver Modules\Question 1

Please list the difference between stub modules and driver modules.

Stub and Driver Modules\Answer 1

Please list the difference between stub modules and driver modules.

No	Stub	Driver
1	Used in Top-Down Integration Testing	Used in Bottom-Up Integration Testing
2	Emulates/Simulates the called function by the target module (i.e., the module under test)	Emulates/Simulates the calling function of the target module (i.e., the module under test)
3	Replace the lower level modules which are not developed	Replace the higher level modules which are not developed

Stub and Driver Modules\Question 2

Bob is assigned to develop the grade retrieving module in a student online transcript system at CUHK. The operating mechanism of the whole system is described below:

(1) A student needs to input her student ID and password to the GUI module.

(2) The GUI module will send the information to the login module to check if the input information is correct.

(3) Once the student information is verified, the login module will send a request with a student ID to the grade retrieving module.

(4) The grade retrieving module will send a request with the student ID to the class schedule module and retrieve the classes enrolled by the student.

(5) The grade retrieving module will call the grade management module with the student ID and the class ID to retrieve the grade that the student got in the class one by one.

(6) The grade retrieving module will call the class information module with the class ID to get the median score of the class.

(7) The grade retrieving module will compare the student's grade with the median score in each class to judge if the performance of the student is good.

(8) The grade retrieving module will send the information about the student's grade, the median score, and whether the performance is good enough of all the classes to the GUI module.

(9) The GUI module will display the related information to the user.

Stub and Driver Modules\Question 2

Bob has finished his work of the grade retrieving module, but all other modules are not finished yet. Now Bob needs to test his module. Please classify all other modules as either stub module, driver module, or not related for Bob's testing, and design the corresponding stub modules and driver modules for Bob to help him finish the testing work.

Stub and Driver Modules\Answer 2

Bob is assigned to develop the grade retrieving module in a student online transcript system at CUHK. The operating mechanism of the whole system is described below:

(1) A student needs to input her student ID and password to the GUI module.

(2) The GUI module will send the information to the login module to check if the input information is correct.

(3) Once the student information is verified, the login module will send a request with a student ID to the grade retrieving module.

(4) The grade retrieving module will send a request with the student ID to the class schedule module and retrieve the classes enrolled by the student.

(5) The grade retrieving module will call the grade management module with the student ID and the class ID to retrieve the grade that the student got in the class one by one.

(6) The grade retrieving module will call the class information module with the class ID to get the median score of the class.

(7) The grade retrieving module will compare the student's grade with the median score in each class to judge if the performance of the student is good.

(8) The grade retrieving module will send the information about the student's grade, the median score, and whether the performance is good enough of all the classes to the GUI module.

(9) The GUI module will display the related information to the user.

Stub and Driver Modules\Answer 2

(1) A student needs to input her student ID and password to the GUI module.

(2) The GUI module will send the information to the login module to check if the input information is correct.

(3) Once the student information is verified, the login module will send a request with a student ID to the grade retrieving module.

(4) The grade retrieving module will send a request with the student ID to the class schedule module and retrieve the classes enrolled by the student.

(5) The grade retrieving module will call the grade management module with the student ID and the class ID to retrieve the grade that the student got in the class one by one.

(6) The grade retrieving module will call the class information module with the class ID to get the median score of the class.

(7) The grade retrieving module will compare the student's grade with the median score in each class to judge if the performance of the student is good.

(8) The grade retrieving module will send the information about the student's grade, the median score, and whether the performance is good enough of all the classes to the GUI module.

(9) The GUI module will display the related information to the user.

Stub and Driver Modules\Answer 2

Driver Modules:

Simulated Login Module: The module can randomly generate a valid student ID and call the grade retrieving module with this student ID.

Stub and Driver Modules\Answer 2

(1) A student needs to input her student ID and password to the GUI module.

(2) The GUI module will send the information to the login module to check if the input information is correct.

(3) Once the student information is verified, the login module will send a request with a student ID to the grade retrieving module.

(4) The grade retrieving module will send a request with the student ID to the class schedule module and retrieve the classes enrolled by the student.

(5) The grade retrieving module will call the grade management module with the student ID and the class ID to retrieve the grade that the student got in the class one by one.

(6) The grade retrieving module will call the class information module with the class ID to get the median score of the class.

(7) The grade retrieving module will compare the student's grade with the median score in each class to judge if the performance of the student is good.

(8) The grade retrieving module will send the information about the student's grade, the median score, and whether the performance is good enough of all the classes to the GUI module.

(9) The GUI module will display the related information to the user.

Stub and Driver Modules\Answer 2

Driver Modules:

Simulated Login Module: The module can randomly generate a valid student ID and call the grade retrieving module with this student ID.

Stub Modules:

Simulated Class Schedule Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a class list from all the classes in the CUHK and return it.

Stub and Driver Modules\Answer 2

(1) A student needs to input her student ID and password to the GUI module.

(2) The GUI module will send the information to the login module to check if the input information is correct.

(3) Once the student information is verified, the login module will send a request with a student ID to the grade retrieving module.

(4) The grade retrieving module will send a request with the student ID to the class schedule module and retrieve the classes enrolled by the student.

(5) The grade retrieving module will call the grade management module with the student ID and the class ID to retrieve the grade that the student got in the class one by one.

(6) The grade retrieving module will call the class information module with the class ID to get the median score of the class.

(7) The grade retrieving module will compare the student's grade with the median score in each class to judge if the performance of the student is good.

(8) The grade retrieving module will send the information about the student's grade, the median score, and whether the performance is good enough of all the classes to the GUI module.

(9) The GUI module will display the related information to the user.

Stub and Driver Modules\Answer 2

Driver Modules:

Simulated Login Module: The module can randomly generate a valid student ID and call the grade retrieving module with this student ID.

Stub Modules:

Simulated Class Schedule Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a class list from all the classes in the CUHK and return it.

Simulated Grade Management Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a score from 0 to 100 and return it.

Stub and Driver Modules\Answer 2

(1) A student needs to input her student ID and password to the GUI module.

(2) The GUI module will send the information to the login module to check if the input information is correct.

(3) Once the student information is verified, the login module will send a request with a student ID to the grade retrieving module.

(4) The grade retrieving module will send a request with the student ID to the class schedule module and retrieve the classes enrolled by the student.

(5) The grade retrieving module will call the grade management module with the student ID and the class ID to retrieve the grade that the student got in the class one by one.

(6) The grade retrieving module will call the class information module with the class ID to get the median score of the class.

(7) The grade retrieving module will compare the student's grade with the median score in each class to judge if the performance of the student is good.

(8) The grade retrieving module will send the information about the student's grade, the median score, and whether the performance is good enough of all the classes to the GUI module.

(9) The GUI module will display the related information to the user.

Stub and Driver Modules\Answer 2

Driver Modules:

Simulated Login Module: The module can randomly generate a valid student ID and call the grade retrieving module with this student ID.

Stub Modules:

Simulated Class Schedule Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a class list from all the classes in the CUHK and return it.

Simulated Grade Management Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a score from 0 to 100 and return it.

Simulated Class Information Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a score from 0 to 100 and return it.

Stub and Driver Modules\Answer 2

(1) A student needs to input her student ID and password to the GUI module.

(2) The GUI module will send the information to the login module to check if the input information is correct.

(3) Once the student information is verified, the login module will send a request with a student ID to the grade retrieving module.

(4) The grade retrieving module will send a request with the student ID to the class schedule module and retrieve the classes enrolled by the student.

(5) The grade retrieving module will call the grade management module with the student ID and the class ID to retrieve the grade that the student got in the class one by one.

(6) The grade retrieving module will call the class information module with the class ID to get the median score of the class.

(7) The grade retrieving module will compare the student's grade with the median score in each class to judge if the performance of the student is good.

(8) The grade retrieving module will send the information about the student's grade, the median score, and whether the performance is good enough of all the classes to the GUI module.

(9) The GUI module will display the related information to the user.

Stub and Driver Modules\Answer 2

Driver Modules:

Simulated Login Module: The module can randomly generate a valid student ID and call the grade retrieving module with this student ID.

Stub Modules:

Simulated Class Schedule Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a class list from all the classes in the CUHK and return it.

Simulated Grade Management Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a score from 0 to 100 and return it.

Simulated Class Information Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a score from 0 to 100 and return it.

Simulated GUI module: The module which has the same API called by the grade retrieving module and can check the validation of the input data format.

Stub and Driver Modules\Answer 2

(1) A student needs to input her student ID and password to the GUI module.

(2) The GUI module will send the information to the login module to check if the input information is correct.

(3) Once the student information is verified, the login module will send a request with a student ID to the grade retrieving module.

(4) The grade retrieving module will send a request with the student ID to the class schedule module and retrieve the classes enrolled by the student.

(5) The grade retrieving module will call the grade management module with the student ID and the class ID to retrieve the grade that the student got in the class one by one.

(6) The grade retrieving module will call the class information module with the class ID to get the median score of the class.

(7) The grade retrieving module will compare the student's grade with the median score in each class to judge if the performance of the student is good.

(8) The grade retrieving module will send the information about the student's grade, the median score, and whether the performance is good enough of all the classes to the GUI module.

(9) The GUI module will display the related information to the user.

Stub and Driver Modules\Answer 2

Driver Modules:

Simulated Login Module: The module can randomly generate a valid student ID and call the grade retrieving module with this student ID

Stub Modules:

Simulated Class Schedule Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a class list from all the classes in the CUHK and return it.

Simulated Grade Management Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a score from 0 to 100 and return it.

Simulated Class Information Module: The module which has the same API called by the grade retrieving module can check the validation of the input data format, randomly generate a score from 0 to 100 and return it.

Simulated GUI module: The module which has the same API called by the grade retrieving module and can check the validation of the input data format.

Thank you!