QUESTION 1 (Theoretical Fundamentals & Fault Failure Model)

Part A

For each of the statements below, state whether you (1) agree, (2) disagree, or (3) both. You must justify your answer.

i. Once you have decided on a test path, it is a simple straightforward process to generate appropriate test data.

ii. If a test requirement can be generated, then it is guaranteed that a test input exists to execute that requirement.

Part B

Answer the following questions based on the code fragment given below:

```
/* This method returns the index of first occurrence of an even
number which is greater than 10
If array is empty, it returns -1. If such value doesn't exist
in the array, it returns -2.
*/

public static int valIndex(int[] numbers){
   if (numbers.length==0)
     return -1;

   for(int i=0;i< numbers.length;i++){
     if((numbers[i]%2==0) || (numbers[i]>10) )
       return i;
  }
   return -2;
 }
```

i. This program contains a fault. What is it? Suggest a correction to the code that will remove the fault.
ii. If possible, give a test case that does not execute the fault. If not, briefly explain why not.
iii. If possible, give a test case that executes the fault, but does not result in an error state. If not, briefly explain why not.
iv. If possible give a test case that results in an error, but not a failure. If not, briefly explain why not.
Hint: Don't forget about the program counter.
v. In the given code, describe the first error state. Be sure to describe the complete state.

QUESTION 2 (Data Flow Coverage & Graph Coverage)

Part A

```
Public void test (int a, int b){  // NODE 1
if (b%2==0)                       // NODE 1
   print "b is even";             // NODE 2
else
   print "b is odd";              // NODE 3
while (b>0){                      // NODE 4
   a++;                           // NODE 5
   b--;                           // NODE 5
}
if (a%2==0)                       // NODE 6
   print "b is even";             // NODE 7
print (a,b)                       // NODE 8
}                                 // NODE 8
```

```
Test Paths
T1: [1,2,4,5,4,6,7,8]  T2:[1,3,4,6,8] T3: [1,2,4,5,4,6,8]
```

a) Draw the Control Flow Graph and mark Defs and Uses of variable a.
b) Identify the sets of Def (a) & Use (a).
c) List down all the DU paths with respect to variable a.
d) Identify the DU paths toured directly & with side trips by each given test path.
e) Identify the minimum test set for 100%, (Direct Tours Only)
    a. All Defs(x) Coverage
    b. All Uses (x) Coverage
    c. All DU Paths coverage

Part B

```
Public void test (int a, int b, int c){
for (int i=1; i<a; i++){
     for (int j=1; j<b; j++){
          if (c%2==0){
               print (i,j);
          else
               break;
     }
}
while (c>0){
     print (C);
     c--;
}
}
```

a) Draw the Control Flow Graph.

b) List down the TR set and Test Set for 100%,
   i. Node Coverage
   ii. Edge Coverage
   iii. Edge Pair Coverage
   iv. Prime Path Coverage

QUESTION 3 (Logic Coverage & FSM)

Part B

Answer the following questions based on the code fragment given below:

```
int foo (int a, int b, int c, int d) {
float e;
if (a == 0) {
  return 0;
}
int x = 5;
if ((a==b) && (c == d) ) {
  x=1;
}
return x;
}
```

Identify the Test Set for 100%,

i. Predicate Coverage
ii. Clause Coverage
iii. Combinatorial Coverage
iv. General Active Clause Coverage

Part B

Answer the following questions based on the deterministic finite state machine below:

Current State	Condition	Next State
Idle	$a \vee b$	Active
Active	$a \wedge b$	Idle
Active	$\neg b$	WindDown
WindDown	a	Idle

i. Draw the finite state machine.
ii. Provide the Test Requirements for GACC for Active to Active Transition.
iii. Provide the Test Requirements for CACC for Active to Idle Transition.