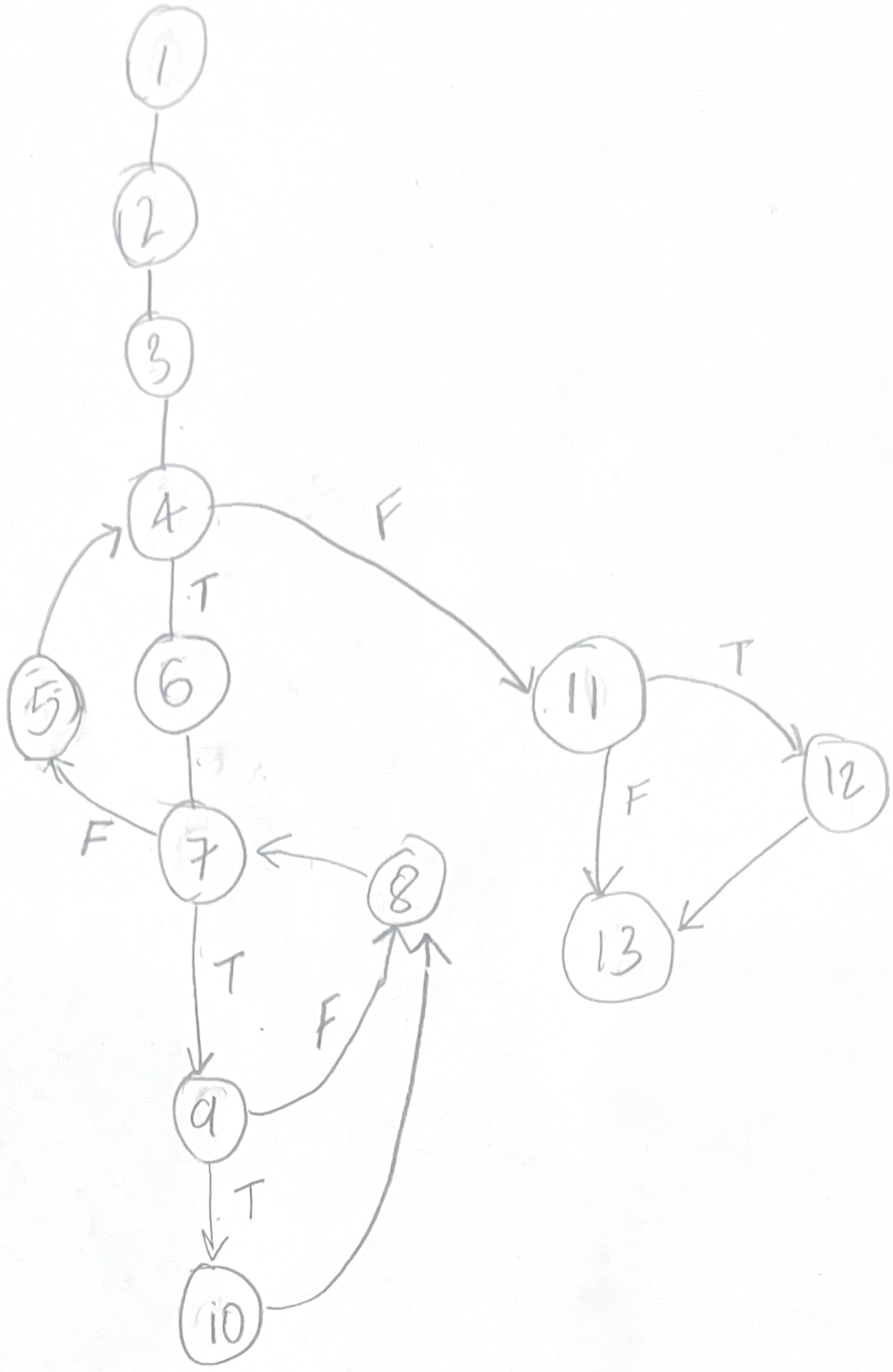
a]



b] E-N+2 = 16-13+2 = 5

1] 1→2→3→4→11→13

2] 1→2→3→4→11→12→13

3] 1→2→3→4→6→7→5→4→11→13

4] 1→2→3→4→6→7→9→8→7→5→4→11→13

5] 1→2→3→4→6→7→9→10→8→7→5→4→11→13

CSE 464 Practice Activity
Control Flow Testing (9/25/25)
10 Points

Name: Bhavya Patel

The findPair method takes an int array and the target int as parameters and It checks all pairs of numbers in the array to see if their sum equals the user's number. If found, it returns the last pair found. If none exists, it returns *"No such pair found."*

```
import java.util.Scanner;
public class PairSumFinder {
    public String findPair(int[] numbers, int target)
     {

        boolean found = false;
        String result = "";

        // Check all possible pairs
        for (int i = 0; i < numbers.length; i++) {

            for (int j = i + 1; j < numbers.length; j++) {

                if (numbers[i] + numbers[j] == target) {

                    result=Pair found: " + numbers[i] + " + " + numbers[j] + " = " + target);
                    found = true;
                }
            }
        }

        if (!found) {
            result = "No such pair found.";
        }

        scanner.close();
        return result;
    }
}
```

a) Draw the flow graph (make sure to label the code clearly)
b) Compute the cyclomatic complexity and then Identify (list) all independent paths
c) Design test cases based on your independent paths above (at least 3)

c] i] numbers = [1, 2, 5], target = 10
ii] numbers = [1, 3, 6], target = 9
iii] numbers = [4, 5], target = 9