

## Assignment-1 PF Theory

**Name:** Deewan Kumar

**Roll No:** 25K-0164

**Section:** BAI-1B

### Part 1 Question No. 1

1. You need to do analysis using PAC, IC, and IPO charts.

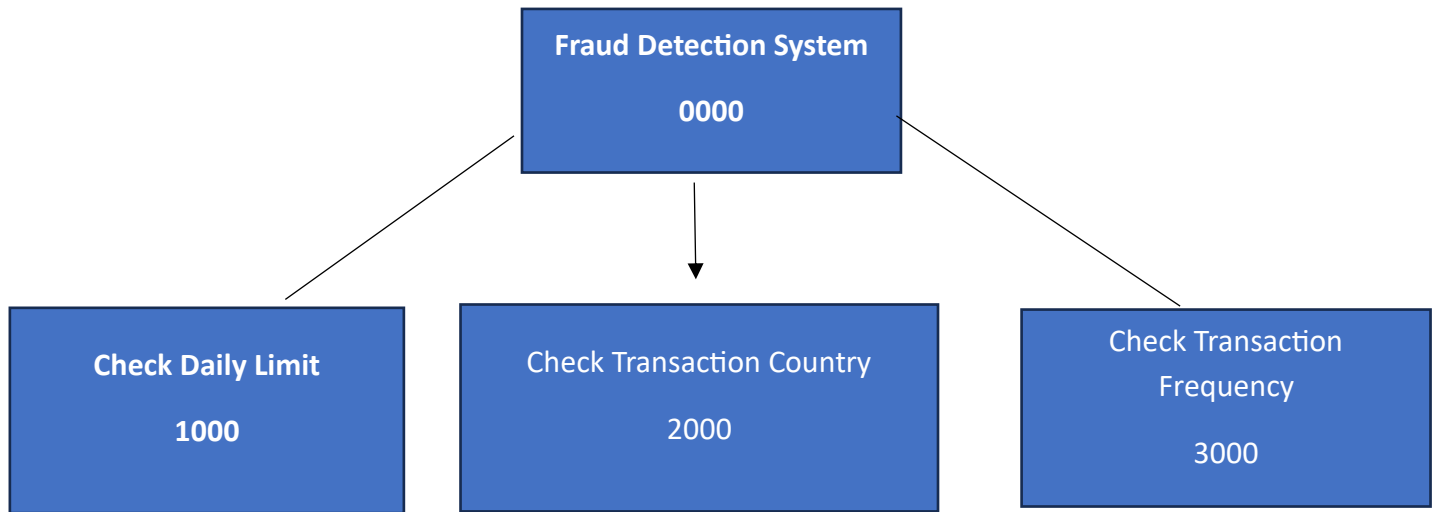
➤ **IPO Chart (Input – Process – Output)**

Input	Process	Output
➤ Daily transaction amount ➤ Country name ➤ Number of transactions	➤ Compare with 5000 ➤ Check against allowed list (Pakistan, UAE) ➤ Compare with 3	➤ Suspicious / Safe ➤ Suspicious/ Safe ➤ Fraud / No fraud

➤ **PAC Chart Problem Analyze Chart**

Given Data	Required Result	Processing Required	Alternative Solution
Daily transaction amount	Check if > 5000	Compare with 5000	Set a lower or higher fixed limit
Transaction country	Check if common or not	Match with Pakistan/UAE	Allow all countries but send alert
Number of transactions	Check if > 3 in short time	Compare with 3	Change limit (e.g., 5 instead of 3)

## "IC" Interactivity Chart



## **Algorithm, Flow Chart, and pseudocode**

### **➤ Algorithm**

1. Start
2. Input daily transaction amount
3. If daily transaction amount > 5000 → Flag as suspicious
4. Else → Print safe
5. Input transaction country name
6. If country is Pakistan or UAE → Print not suspicious
7. Else → Print suspicious
8. Input number of transactions in a short time
9. If number of transactions > 3 → Print fraud transaction
10. Else → Print no fraud
11. End

### **➤ Pseudocode**

BEGIN

INPUT daily\_Limit

IF daily\_Limit > 5000 THEN

    PRINT "Transaction should be flagged"

ELSE

    PRINT "Transaction is safe"

ENDIF

INPUT foreign\_Country

IF foreign\_Country = "Pakistan" OR foreign\_Country = "UAE" THEN

    PRINT "Transaction is not suspicious"

ELSE

    PRINT "Transaction is suspicious"

ENDIF

INPUT number\_Transaction

IF number\_Transaction > 3 THEN

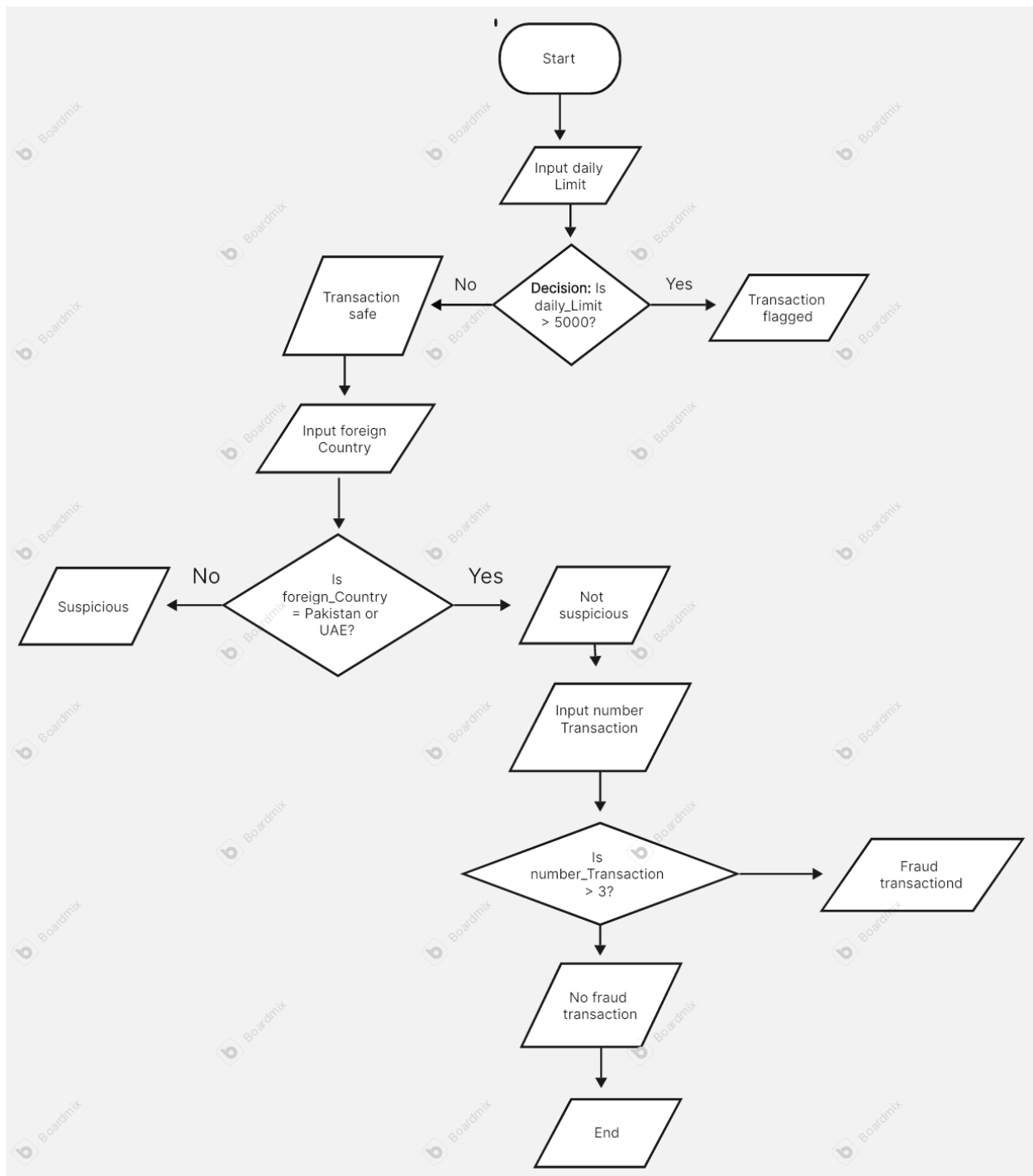
    PRINT "It is sign of fraud transaction"

ELSE

    PRINT "No fraud transaction"

ENDIF

END



**Flow Chart**

## Question No. 2

### ➤ PAC Chart

Given Data	Required Result	Processing Required	Alternative Solution
X-coordinate, Y-coordinate	Identify which quadrant the point lies in	Check signs of x and y to determine quadrant	Use a formula or mapping table instead of multiple conditions

### ➤ IPO Chart

Input	Process	Output
X and Y coordinates	Check conditions: (+,+), (-,+), (-,-), (+,-)	Print which quadrant the point lies in

### ➤ Algorithm

#### Algorithm (in words)

1. Start the program.
2. Ask the user to enter the value of the x-coordinate.
3. Ask the user to enter the value of the y-coordinate.
4. Check the signs of x and y:

- If x is positive and y is positive, then the point lies in Quadrant I.
- If x is negative and y is positive, then the point lies in Quadrant II.
- If x is negative and y is negative, then the point lies in Quadrant III.
- If x is positive and y is negative, then the point lies in Quadrant IV.

5. Display the quadrant result to the user.

6. End the program.

### ➤ Pseudocode

BEGIN

PRINT "This program will identify the quadrant of a point"

INPUT x\_coordinate

INPUT y\_coordinate

IF x\_coordinate > 0 AND y\_coordinate > 0 THEN

PRINT "The point (x, y) lies in Quadrant I"

ENDIF

IF x\_coordinate < 0 AND y\_coordinate > 0 THEN

PRINT "The point (x, y) lies in Quadrant II"

ENDIF

IF x\_coordinate < 0 AND y\_coordinate < 0 THEN

PRINT "The point (x, y) lies in Quadrant III"

ENDIF

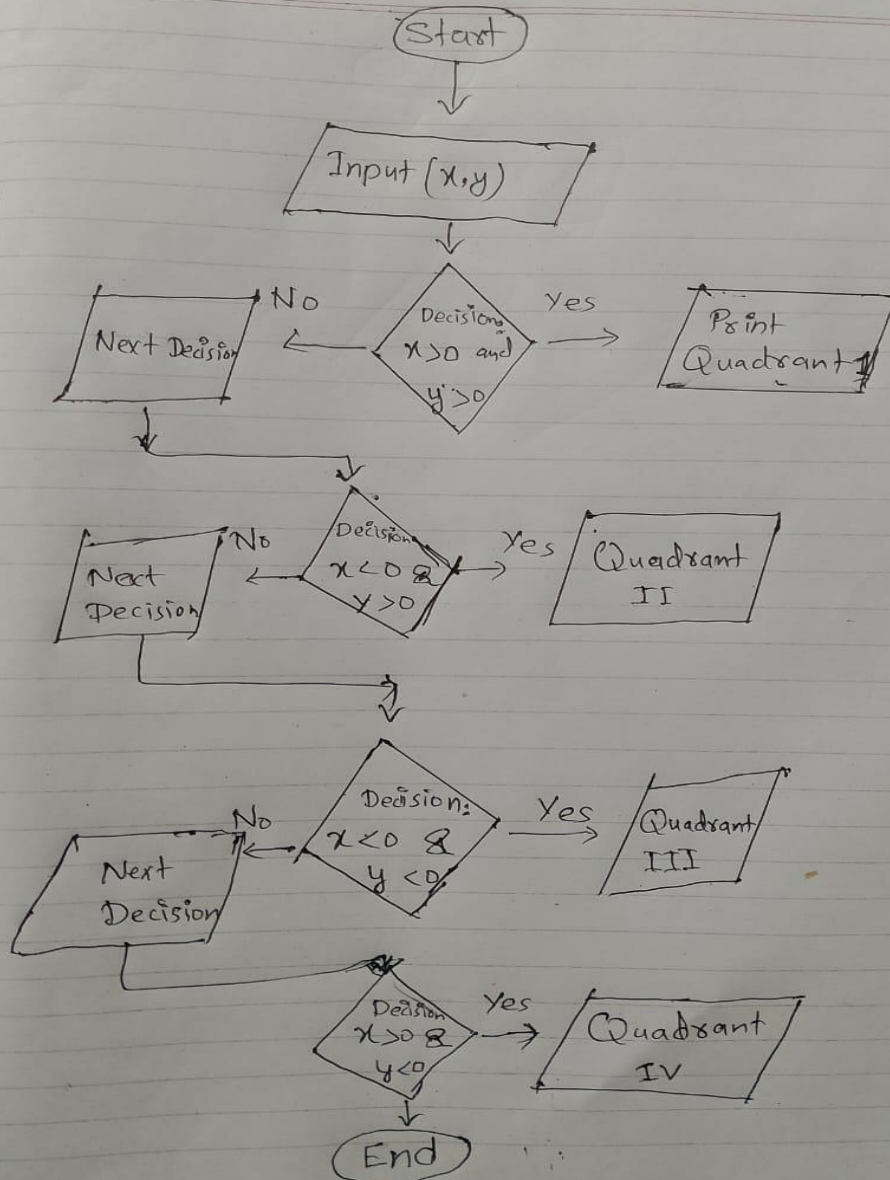
IF x\_coordinate > 0 AND y\_coordinate < 0 THEN

PRINT "The point (x, y) lies in Quadrant IV"

ENDIF

END

Question: 2.





### Question No. 3

#### ➤ IPO Chart

Input	Process	Output
Age, eyesight result, written test result, driving test result, medical certificate	Step-by-step checks (age $\geq 18$ $\rightarrow$ eyesight $\rightarrow$ written $\rightarrow$ driving $\rightarrow$ if age $> 60 \rightarrow$ medical certificate)	Eligible or Not Eligible

#### PAC Chart

Given Data	Required Result	Processing Required	Alternative Solution
<ul style="list-style-type: none"><li>• Age of applicant</li><li>• Eyesight test result (P/F)</li><li>• Written test result (P/F)</li><li>• Driving test result (P/F)</li><li>• Medical certificate (Y/N)</li></ul>	<ul style="list-style-type: none"><li>• Check if applicant meets minimum age (18)</li><li>• Verify eyesight condition</li><li>• Verify written test</li><li>• Verify driving skill</li><li>• Verify fitness if age <math>&gt; 60</math></li></ul>	<ul style="list-style-type: none"><li>• Compare with 18</li><li>• If Passed <math>\rightarrow</math> continue, else <math>\rightarrow</math> reject</li><li>• If Passed <math>\rightarrow</math> continue, else <math>\rightarrow</math> reject</li><li>• If Passed <math>\rightarrow</math> check age, else <math>\rightarrow</math> reject</li><li>• If Yes <math>\rightarrow</math> eligible, else <math>\rightarrow</math> not eligible</li></ul>	<ul style="list-style-type: none"><li>• Store minimum age in a variable</li><li>• Accept medical report instead</li><li>• Online written test</li><li>• Simulator-based driving check</li><li>• Exempt older applicants with history</li></ul>

### ➤ Algorithm

1. Start the program.
2. Ask the applicant for their age.
3. If the age is less than eighteen, then display "You are not eligible".
4. Otherwise, if the age is eighteen or greater, then:
  - Ask if the applicant passed the eyesight exam.
  - If the eyesight exam result is Fail, then display "You need eye glasses".
  - Otherwise, if the eyesight exam result is Pass, then:
    - Ask if the applicant passed the written test.
    - If the written test result is Fail, then display "Ineligible for driving license".
    - Otherwise, if the written test result is Pass, then:
      - Ask if the applicant passed the driving test.
      - If the driving test result is Fail, then display "Ineligible for driving license".
      - Otherwise, if the driving test result is Pass, then:
        - If the age is less than sixty, then display "You are eligible".
        - Otherwise, if the age is sixty or greater, then:
          - Ask if the applicant has a medical certificate.
          - If the answer is Yes, then display "You are eligible".
          - Otherwise, if the answer is No, then display "You are not eligible".
5. End the program

### ➤ Pseudocode

BEGIN

PRINT "Driving License Eligibility Check"

INPUT age

IF age >= 18 THEN

PRINT "Eligible and Ready for Next Question"

```
INPUT eyesight_exam

IF eyesight_exam = 'P' THEN

    PRINT "Move to next Question"

    INPUT written_exam

    IF written_exam = 'P' THEN

        PRINT "Move to next Question"

        INPUT driving_test

        IF driving_test = 'P' THEN

            IF age < 60 THEN

                PRINT "You are eligible"

            ELSE

                PRINT "Go for next Question"

                INPUT medical_certificate

                IF medical_certificate = 'Y' OR 'y' THEN

                    PRINT "You are eligible"

                ELSE

                    PRINT "You are not eligible"

                ENDIF

            ENDIF

        ENDIF

    ELSE

        PRINT "Ineligible for Driving license"

    ENDIF

ELSE

    PRINT "Ineligible for Driving license"

ENDIF

ELSE

    PRINT "You need eye glasses"

ENDIF

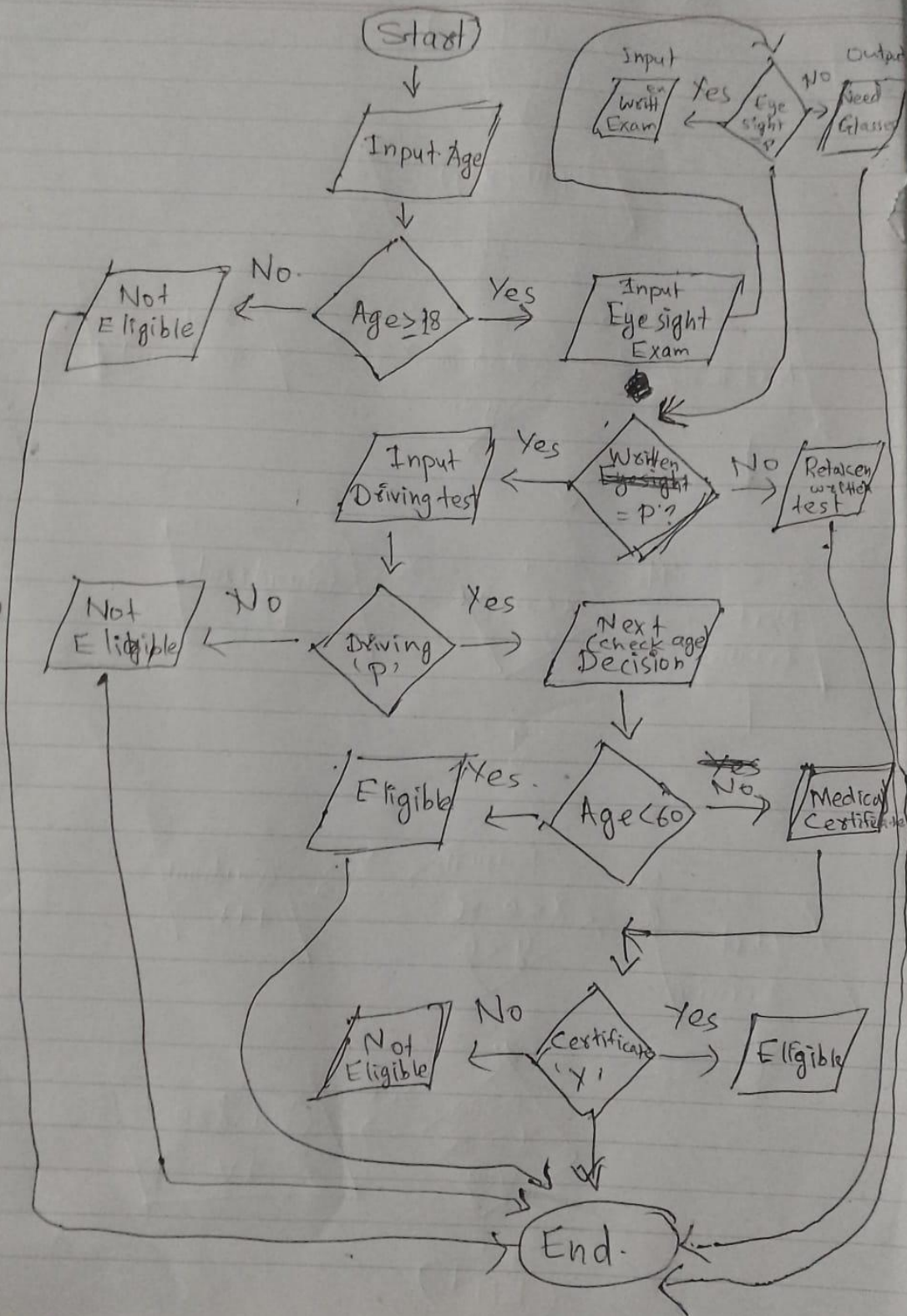
ELSE
```

```
PRINT "You are not eligible"
```

```
ENDIF
```

```
END
```

Question:3.



#### Question-4

➤ IPO Chart

Input	Process	Output
5 integers (card values)	Check if exactly three are the same and two are the same but different from the three	Display result whether it is a Full House or not

➤ PAC Chart

Given Data	Required Result	Processing Required	Alternative Solution
<ul style="list-style-type: none"><li>5 integers (cards between 1–13)</li></ul>	<ul style="list-style-type: none"><li>Identify if the hand is a Full House or not</li></ul>	<ul style="list-style-type: none"><li>Compare cards to check if three are of one rank and two of another</li></ul>	<ul style="list-style-type: none"><li>Use arrays and frequency count of each card</li></ul>

➤ **Algorithm**

1. Start the program
2. Ask the user to enter five integers between one and thirteen.
3. Check all possible combinations:
  - If three cards have the same rank and the other two cards have the same rank (different from the first three), then it is a full house.
4. If the condition is true, display "The hand is a Full House".
5. Otherwise, display "The hand is NOT a Full House".
6. End the program.

➤ **Pseudocode**

Start

Input card1, card2, card3, card4, card5

If (three cards same AND other two cards same AND both groups different) then

    Print "The hand is a Full House"

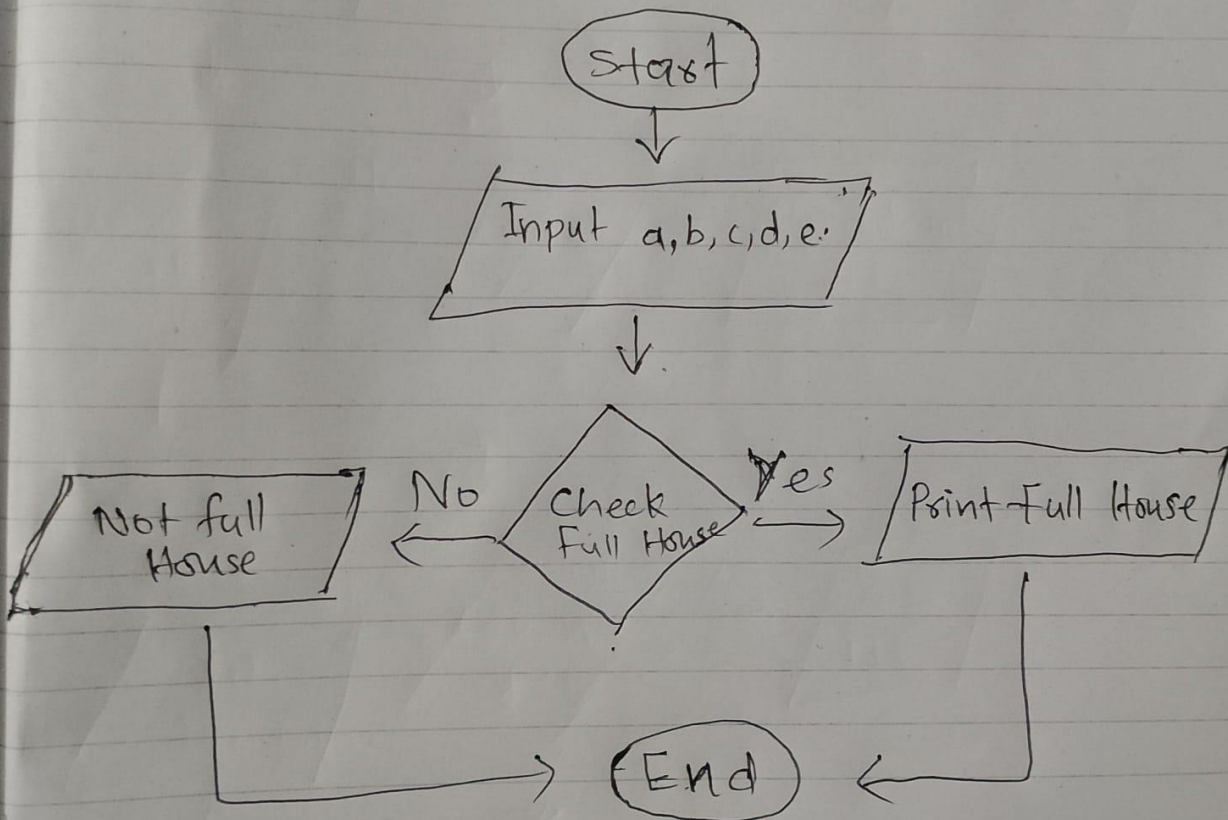
Else

    Print "The hand is NOT a Full House"

End If

Stop

Question: 4.





## Question No. 5

### ➤ PAC Analysis (Problem, Analysis, Chart)

Given Data	Required Result	Processing Required	Alternative Solution
<ul style="list-style-type: none"><li>• User can input integers (0–9) repeatedly</li><li>• User may enter invalid numbers (&lt;0 or &gt;9)</li><li>• Multiple inputs allowed</li><li>• Numbers are integers</li></ul>	<ul style="list-style-type: none"><li>• Count how many times each number (0–9) is entered</li><li>• Stop input immediately when invalid number is entered</li><li>• Display table of numbers and their counts</li><li>• Show output in a clean tabular format</li></ul>	<ol style="list-style-type: none"><li>1. Initialize array count[10] = {0}</li><li>2. Read number from user</li><li>3. If number is 0–9, increment count[num]</li><li>4. Repeat until invalid input</li></ol> <ul style="list-style-type: none"><li>• Check `if(num &lt; 0)</li><li>• Loop from 0 to 9 and print each number and count[i]</li><li>• Format output neatly using printf and spacing</li></ul>	<ul style="list-style-type: none"><li>• Use switch statements for each number instead of an array</li><li>• Print formatted output from file instead of console</li><li>• Use %-6d formatting in printf for alignment</li></ul>

➤ **IPO Chart**

Input	Process	Output
Integers entered by user	<ul style="list-style-type: none"><li>- Check if input is 0-9</li><li>- Increment corresponding count element</li><li>- Stop loop on invalid input</li></ul>	Table: Number Number of Occurrences 0-9 with counts

➤ **Algorithm (Step-by-Step)**

1. Start
2. Initialize an integer array count[10] with all elements 0
3. Repeat indefinitely:
  - a. Ask the user to "Enter numbers:"
  - b. Read num from the user
  - c. If num is between 0 and 9 (inclusive), increment count[num]
  - d. Otherwise, exit the loop
4. Print the header: "Number of Occurrences"
5. For each i from 0 to 9:
  - a. Print i and count[i]
6. End

➤ **Pseudocode**

BEGIN

DECLARE count[10] AS INTEGER

```
INITIALIZE count[0..9] TO 0
```

```
WHILE TRUE
```

```
    PRINT "Enter numbers:"
```

```
    READ num
```

```
    IF num >= 0 AND num <= 9 THEN
```

```
        count[num] = count[num] + 1
```

```
    ELSE
```

```
        BREAK
```

```
    ENDIF
```

```
ENDWHILE
```

```
PRINT "Number of Occurrences"
```

```
FOR i = 0 TO 9
```

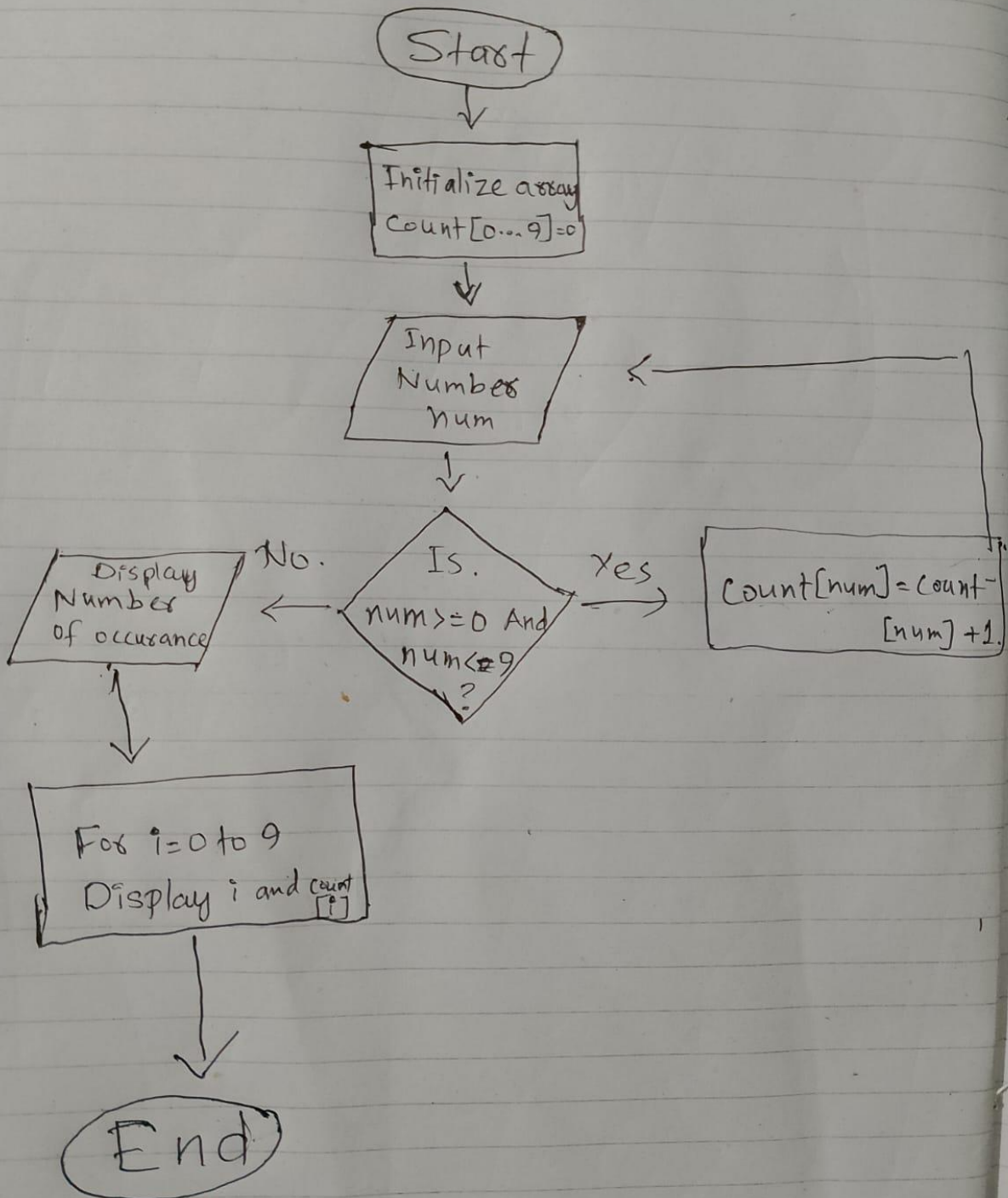
```
    PRINT i, count[i]
```

```
ENDFOR
```

```
END
```

➤ **Flow Chart**

## Question-5-A



### Question-6

Draw and Run of

1	Condition Checked	Result	Action	Values (i, j, k)
	Initial Values	-	-	3, 5, 2007
2	if (3 < 5)	TRUE	Enter first if block	3, 5, 2007
3	if (5 < 7)	TRUE	i = j;	5, 5, 2007