Question 4:

PAC Chart:

DATA GIVEN	REQUIRED RESULT(S)
• The ranks of cards in a poker hand.	• A clear yes or no answer as to whether the hand is a Full House.
REQUIRED PROCESSING	**SOLUTION ALTERNATIVE(S)**
• Take the ranks of each card in the hand as input and store them in individual variables. • Use two variables to store the two unique ranks required for a Full House (one for the three-of-a-kind and one for the pair). • Use two additional variables, initialized to zero, to count how many cards of each rank are in the hand. • The first card entered establishes the first rank. The count for this rank starts at one. • The second card is compared to the first. If it's the same rank, the count for the first rank increases. If it's a different rank, it becomes the second unique rank, and its count starts at one. • The remaining cards are compared to the first and second ranks. If a card matches an existing rank, that rank's counter is incremented. If it doesn't match either of the two existing ranks, the hand cannot be a Full House. • Finally, check if one rank has a count of three and the other has a count of two. If this condition is met, the hand is a Full House. Otherwise, it is not.	• A better approach is to not make "piles" of cards but to simply compare the ranks of the cards individually to determine the two unique ranks and their counts. • The processing should check for counts of 3 and 2 at the end, which confirms the Full House.

IPO Chart:

INPUT	PROCESS	MODULE REF	OUTPUT
• Card1 • Card2 • Card3 • Card4 • Card5	• Prompt the user to enter the rank of each card one at a time. • Take the five ranks as input and store them in individual variables. • Use two variables (rank1, rank2) to store the two distinct ranks you're looking for. • Use two counter variables (rank1Count, rank2Count), initialized to 0, to track how many of each rank you find. • Store the rank of the first card in rank1 and increment rank1Count to 1. • Check if the second card's rank is the same as rank1. If so, increment rank1Count. Otherwise, store its rank in rank2 and increment rank2Count to 1. • For the remaining cards (Card3 through Card5), check their rank against rank1 and rank2. If a match is found, increment the corresponding counter. If not, the hand cannot be a Full House. • Finally, check if the counts are rank1Count = 3 and rank2Count = 2, or rank1Count = 2 and rank2Count = 3. • Output "Hand is Full House" if the condition is met. • Otherwise, output "Hand is not Full House".	PRINT INPUT COMPUTE COMPUTE IF-THEN IF-ELSE IF-THEN OUTPUT OUTPUT	"Hand is Full House" or "Hand is not Full House."

Algorithm:

1. Request the user to provide the ranks of the five cards in their hand.
2. Receive each rank as input and assign it to a distinct variable.
3. Declare two variables to hold the unique ranks (rank1, rank2) and two separate integer variables to track their counts (rank1Count, rank2Count), initializing both counters to zero.
4. Set rank1 equal to the rank of the first card, then increment rank1Count to 1.
5. Examine the second card. If its rank is the same as rank1, increment rank1Count. Otherwise, set rank2 to its rank and increment rank2Count to 1.
6. For the third card, if it matches rank1, increment rank1Count. If not, and if rank2Count is still zero, set rank2 to the third card's rank and increment rank2Count. If rank2 already holds a value, check if the third card matches rank2 and, if so, increment rank2Count.
7. Repeat the process in Step 6 for the fourth and fifth cards.
8. Verify the final counts. The hand is a Full House if rank1Count is 3 and rank2Count is 2, or if the counts are the reverse (rank1Count is 2 and rank2Count is 3).
9. Display "Full House" if the condition in Step 8 is true; otherwise, display "Not a Full House".

Pseudo Code:

```
1. START
2. SET rank1Count to 0
3. SET rank2Count to 0
4. PRINT "Enter the ranks of your Poker Hand one by one"
5. PRINT "Rank of Card 1:"
6. INPUT card1
7. PRINT "Rank of Card 2:"
8. INPUT card2
9. PRINT "Rank of Card 3:"
10. INPUT card3
11. PRINT "Rank of Card 4:"
12. INPUT card4
13. PRINT "Rank of Card 5:"
14. INPUT card5
15. SET rank1 = card1
16. INCREMENT rank1Count
17. IF card2 = rank1 THEN
18.     INCREMENT rank1Count
19. ELSE
20.     SET rank2 = card2
21.     INCREMENT rank2Count
22. ENDIF
23. IF card3 = rank1 THEN
```

```
    INCREMENT rank1Count
ELSE
    IF rank2Count = 0 THEN
        SET rank2 = card3
        INCREMENT rank2Count
    ELSE IF card3 = rank2 THEN
        INCREMENT rank2Count
    ENDIF
ENDIF
IF card4 = rank1 THEN
    INCREMENT rank1Count
ELSE
    IF rank2Count = 0 THEN
        SET rank2 = card4
        INCREMENT rank2Count
    ELSE IF card4 = rank2 THEN
        INCREMENT rank2Count
    ENDIF
ENDIF
IF card5 = rank1 THEN
    INCREMENT rank1Count
ELSE
    IF rank2Count = 0 THEN
        SET rank2 = card5
        INCREMENT rank2Count
    ELSE IF card5 = rank2 THEN
        INCREMENT rank2Count
    ENDIF
ENDIF
IF (rank1Count=3 AND rank2Count=2) OR (rank1Count=2 AND rank2Count=3) THEN
    PRINT "This is a Poker Full Hand!"
ELSE
    PRINT "This is not a Poker Full Hand!"
ENDIF
END
```