PF Assignment #1

Name: Taher Mustansir

Roll: 25K-0119

All code in this file is available at:
https://github.com/TaherMustansir1929/bai-1c-pf-assignment-1

Question #1:

PAC Chart:

Given Data

Transaction Details:

- Amount
- country
- timestamp

Customer Info:

- Fixed daily spending limit.
- Customer's total spending for the current day
- List of countries where the customer commonly shops
- List of the customer's recent transactions (within the last hour)
- Transaction frequency limit (more than 3 transactions in one hour)

Required Result

- Transactions flagged either “NORMAL” or “SUSPICIOUS.”
- Updated customer profile.

Processing Required

1. READ Transaction Details & Customer Info
2. Check daily spending limit
3. Check for foreign country
4. Check for transaction magnitude within the last hour
5. Apply if/else statement to filter out suspicious transactions

Solution Alternatives

Sequential Checks:

- Nested if/else checks
- Individual if checks

Early exit:

6. Raise/flag the suspicious transactions for further investigation.
7. Update customer info after saving transaction details

- IF/else ladder (exit after any one condition is met)

OOP:

- Use class system for handling transactions and customer info

IPO Chart:

Input	Processing	Module Reference	Output
Transaction Details: ● Amount ● country ● timestamp.	1. Enter Amount 2. Enter Country 3. Enter Timestamp 4. Enter Customer Info 5. Check Total_Amount <= Spending Limit 6. Check Common_Countries includes Country 7. Check Total_Transactions_Per_Hour <= 3 8. Flag transactions as “Normal” or “Suspicious” 9. Display if a transaction is	1. READ 2. READ 3. READ 4. READ 5. SELECTION 6. SELECTION 7. SELECTION 8. COMPUTE 9. PRINT 10. END	**Transaction status:** ● Normal or ● Suspicious

flagged
“Suspicious”
10. End

Algorithm:

1. **START**
2. **Initialize** predefined variables:
    - Spending_Limit = 5000
    - Common_Countries = {“Pakistan”, “UAE”}
    - Max_Transactions_Per_Hour = 3
3. **Declare** customer info:
    - Total_Transactions_Per_Hour (integer)
    - Total_Spending_Today (integer)
4. **READ** transaction details:
    - Amount
    - Country
    - Timestamp
5. **READ** customer info:
    - Total_Transactions_Per_Hour
    - Total_Spending_Today
6. **Declare** a flag as is_suspicious = false
7. **Set** Total_Spending_Today = Total_Spending_Today + Amount
8. **Increment** Total_Transactions_Per_Hour by one
9. **Check** for suspicious conditions:

    a. **IF** Total_Spending_Today > Spending_Limit:

        i. **SET** is_suspicious to TRUE.

        ii. **OUTPUT** "Suspicious: Daily spending limit exceeded."

    b. **ELSE IF** the Country of the transaction is NOT in Common_Countries:

        i. **SET** is_suspicious to TRUE.

        ii. **OUTPUT** "Suspicious: Foreign country transaction."

    c. **ELSE IF** the number of Total_Transactions_Per_Hour is >= Max_Transactions_Per_Hour:

        i. **SET** is_suspicious to TRUE.

        ii. **OUTPUT** "Suspicious: Too many transactions in a short period."

10. **Check** if is_suspicious is True:

i. **OUTPUT** “Transaction is Normal”

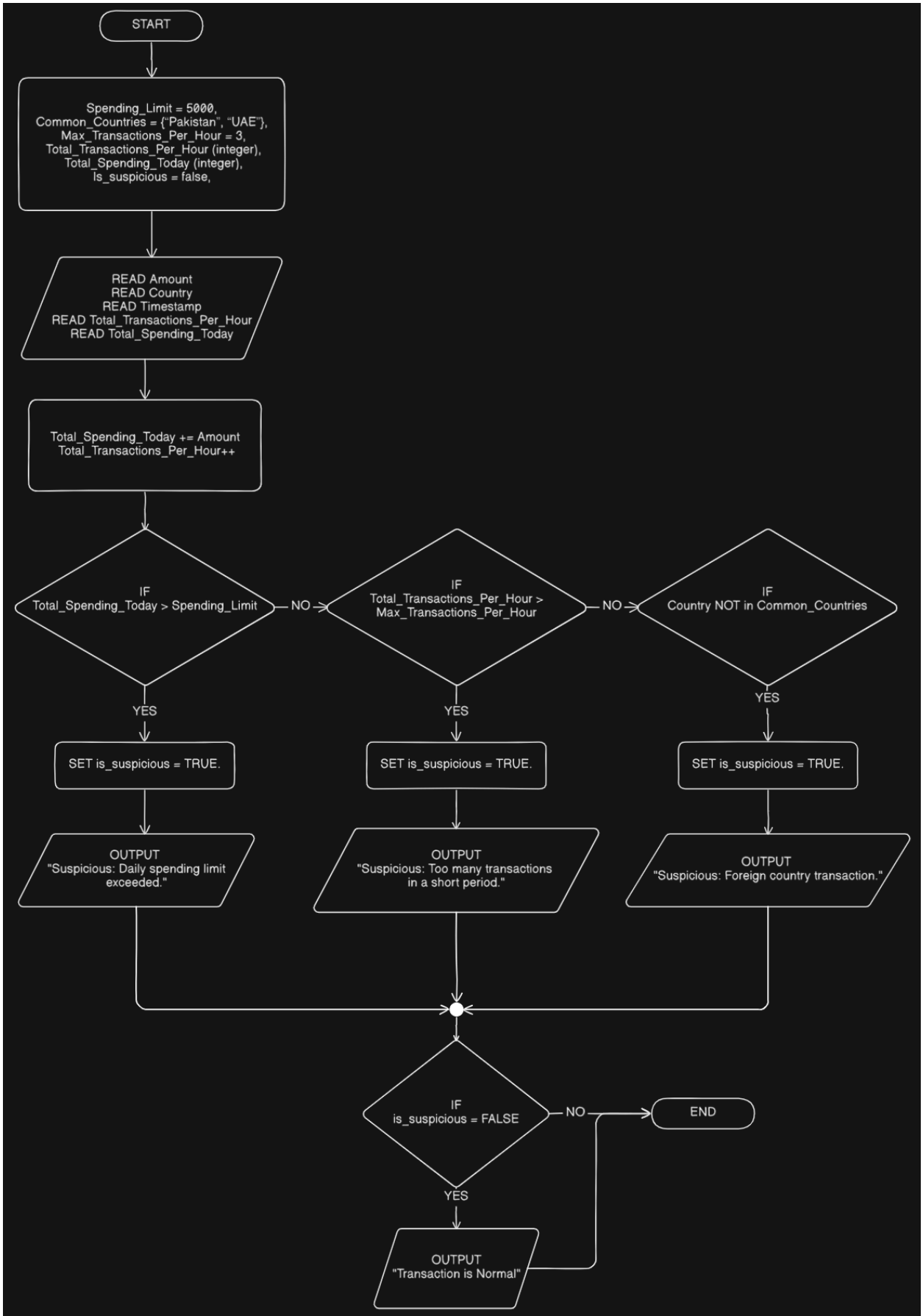
11. **END**

Pseudocode:

```
START
DECLARE
        Spending_Limit = 5000,
        Common_Countries = {“Pakistan”, “UAE”},
         Max_Transactions_Per_Hour = 3,
         Total_Transactions_Per_Hour (integer),
         Total_Spending_Today (integer),
        Is_suspicious = false,
READ
        Amount
        Country
        Timestamp
        Total_Transactions_Per_Hour
        Total_Spending_Today
COMPUTE
        Total_Spending_Today += Amount
        Total_Transactions_Per_Hour++
IF Total_Spending_Today > Spending_Limit:
        SET is_suspicious = TRUE.
        OUTPUT "Suspicious: Daily spending limit exceeded."
ELSE IF Country NOT in Common_Countries:
        SET is_suspicious = TRUE.
```

```
        OUTPUT "Suspicious: Foreign country transaction."
ELSE IF Total_Transactions_Per_Hour > Max_Transactions_Per_Hour:
        SET is_suspicious = TRUE.
        OUTPUT "Suspicious: Too many transactions in a short period."
ENDIF
IF is_suspicious = FALSE:
        OUTPUT “Transaction is Normal”
ENDIF
END
```

Flowchart: (next page) (link: https://app.eraser.io/workspace/r36Dc8la4TuC2qSB4vY3?origin=share)


START
Spending_Limit = 5000,
Common_Countries = {"Pakistan", "UAE"},
Max_Transactions_Per_Hour = 3,
Total_Transactions_Per_Hour (integer),
Total_Spending_Today (integer),
Is_suspicious = false,
READ Amount
READ Country
READ Timestamp
READ Total_Transactions_Per_Hour
READ Total_Spending_Today
Total_Spending_Today += Amount
Total_Transactions_Per_Hour++
IF
Total_Spending_Today > Spending_Limit
NO
IF
Total_Transactions_Per_Hour >
Max_Transactions_Per_Hour
NO
IF
Country NOT in Common_Countries
YES
SET is_suspicious = TRUE.
OUTPUT
"Suspicious: Daily spending limit
exceeded."
YES
SET is_suspicious = TRUE.
OUTPUT
"Suspicious: Too many transactions
in a short period."
YES
SET is_suspicious = TRUE.
OUTPUT
"Suspicious: Foreign country transaction."
IF
is_suspicious = FALSE
NO
END
YES
OUTPUT
"Transaction is Normal"

C Code:

```
#include <stdbool.h>
#include <stdio.h>
#include <string.h>

int main() {
  const int spending_limit = 5000;
  const char *common_countries[] = {"pakistan", "uae"};
  const int max_transactions_per_hour = 3;
  int total_transactions_per_hour;
  int total_spending_today;
  bool is_suspicious = false;

  int amount;
  char country[50];

  printf("Enter transaction amount: ");
  scanf("%d", &amount);
  printf("Enter transaction country: ");
  scanf("%s", country);
  printf("Enter total transactions per hour before this one: ");
  scanf("%d", &total_transactions_per_hour);
  printf("Enter total spending today before this one: ");
  scanf("%d", &total_spending_today);

  total_spending_today += amount;
  total_transactions_per_hour++;

  if (total_spending_today > spending_limit) {
    is_suspicious = true;
    printf("Suspicious: Daily spending limit exceeded.\n");
  } else {
    bool is_common_country = false;
    for (int i = 0; i < 2; i++) {
      if (strcmp(country, common_countries[i]) == 0) {
        is_common_country = true;
        break;
      }
    }
    if (!is_common_country) {
      is_suspicious = true;
      printf("Suspicious: Foreign country transaction.\n");
    } else if (total_transactions_per_hour >
max_transactions_per_hour) {
      is_suspicious = true;
```

```
      printf("Suspicious: Too many transactions in a short
period.\n");
    }
  }

  if (!is_suspicious) {
    printf("Transaction is Normal\n");
  }

  return 0;
}
```

Question #2:

PAC Chart:

Given Data	Required Result
• X (value of the coordinate on x-axis) • Y (value of the coordinate on y-axis)	Quadrant in which the coordinate lies

Processing Required	Solution Alternatives
Check if X > 0 & Y > 0 Then 1st Quadrant Else Check if X < 0 & Y > 0 Then 2nd Quadrant Else Check if X < 0 & Y < 0 Then 3rd Quadrant Else Check if X > 0 & Y < 0 Then 4th Quadrant	• Nested if-else • Switch case with objects/dictionary • Double if checks with logical operators • Skip last if statement

IPO Chart:

Input	Processing	Module Reference	Output
X (x-axis value) Y (y-axis value)	1. Enter X 2. Enter Y 3. Check X>0 & Y>0	1. READ 2. READ 3. SELECTION	The number of quadrant with respect

4. Check X<0 & Y>0	4. SELECTION	to the coordinate
5. Check X<0 & Y<0	5. SELECTION	points
6. Check X>0 & Y<0	6. SELECTION	
7. Display the quadrant number	7. OUTPUT	
8. End	8. END	

Algorithm:

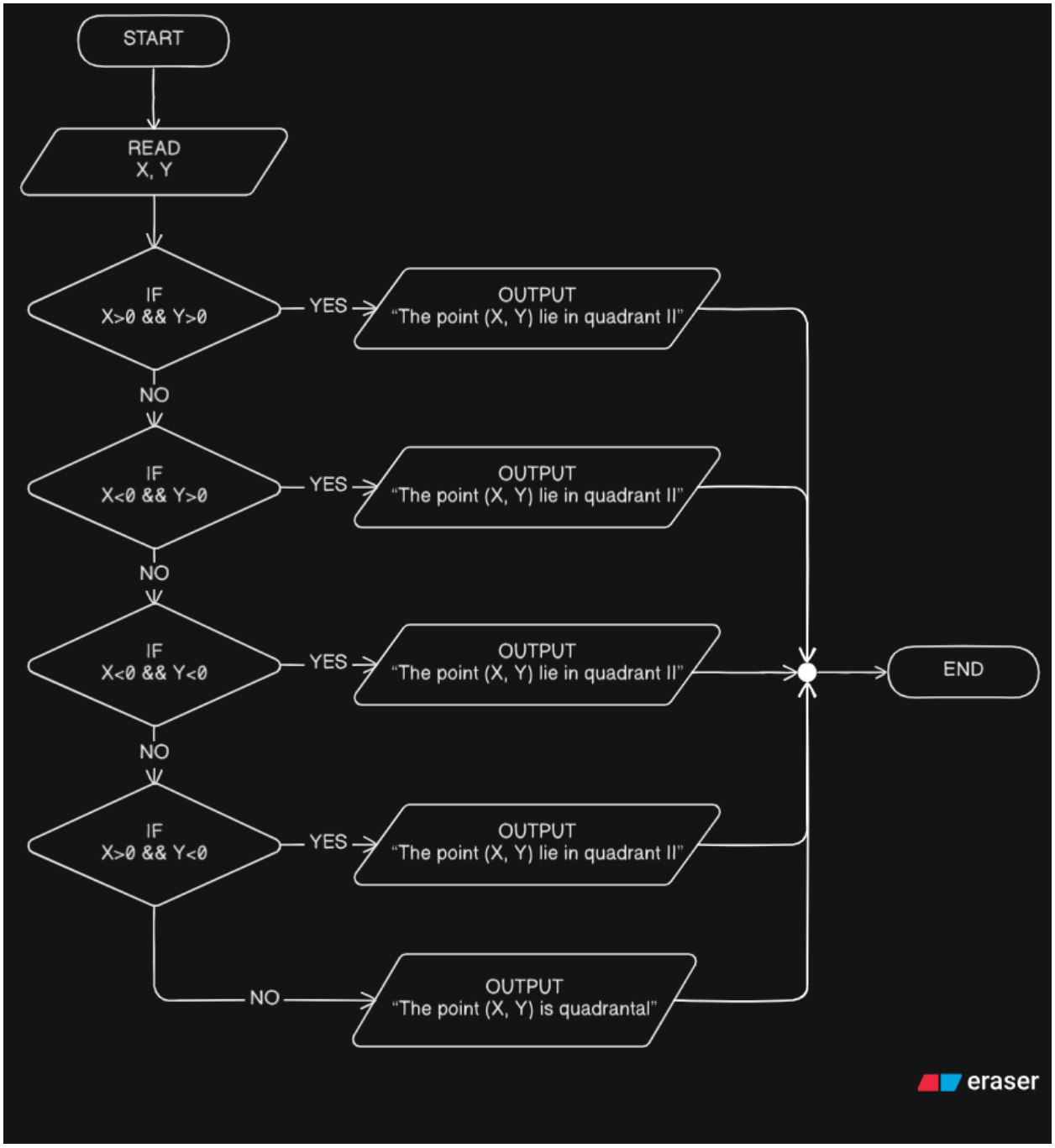
1. Start
2. Declare two variables for both axis:
    a. X
    b. Y
3. Read the values of X & Y from the user
4. Check if X > 0 and Y > 0 Then Display “The point (X, Y) lie in quadrant I” and exit otherwise continue...
5. Check if X < 0 and Y > 0 Then Display “The point (X, Y) lie in quadrant II” and exit otherwise continue...
6. Check if X < 0 and Y < 0 Then Display “The point (X, Y) lie in quadrant III” and exit otherwise continue...
7. Check if X > 0 and Y < 0 Then Display “The point (X, Y) lie in quadrant IV” and exit otherwise continue...
8. If no condition matches Then Display “The point (X, Y) is quadrantal”
9. End

Pseudocode:

```
START

DECLARE

    X (number) , Y (number)

READ X & Y

IF X > 0 && Y > 0:

    OUTPUT “The point (X, Y) lie in quadrant I”
```

```
ELSE IF X < 0 && Y > 0:
        OUTPUT “The point (X, Y) lie in quadrant II”
ELSE IF X < 0 && Y < 0:
        OUTPUT “The point (X, Y) lie in quadrant III”
ELSE IF X > 0 && Y < 0:
        OUTPUT “The point (X, Y) lie in quadrant IV”
ELSE
        OUTPUT “The point (X, Y) is quadrantal”
ENDIF
END
```

Flowchart: (next page) (link: https://app.eraser.io/workspace/WGxYm9ZMbisNRrd78N80?origin=share)



C Code:

```
#include <stdio.h>

int main() {
  int x, y;
  printf("Enter the value of coordinate (x, y): ");
```

```
  scanf("%d %d", &x, &y);

  if (x > 0 && y > 0) {
    printf("The point (%d, %d) lies in quadrant I\n", x, y);
  } else if (x < 0 && y > 0) {
    printf("The point (%d, %d) lies in quadrant II\n", x, y);
  } else if (x < 0 && y < 0) {
    printf("The point (%d, %d) lies in quadrant III\n", x, y);
  } else if (x > 0 && y < 0) {
    printf("The point (%d, %d) lies in quadrant IV\n", x, y);
  } else {
    printf("The point (%d, %d) is quadrantal\n", x, y);
  }
}
```

Question #2:

PAC Chart:

Given Data	Required Result
● Age ● Eyesight Test ● Written Test ● Driving Test ● Medical Fitness Certificate	Whether the user is eligible for driving license or not
Processing Required	**Solution Alternatives**
Mark as ineligible IF: ● Age >= 18 ● Passed eyesight test ● Passed written test ● Passed driving test ● Age < 60 ● Has medical fitness certificate	● Use nested if-else ● Use switch case ● Use OOP classes

IPO Chart:

Input	Processing	Module Reference	Output
Age Eyesight Test Written Test Driving Test Medical Fitness Certificate	1. Enter Age 2. Check IF Age < 18 Then STOP 3. Enter Eyesight_Test 4. Check IF Eyesight_Test = failed Then STOP 5. Enter Written_Test 6. Check IF Written_Test = failed Then STOP 7. Enter Driving_Test 8. Check IF Driving_Test = failed Then STOP 1. Enter Medical_fitness_certificate 2. Check IF Medical_fitness_certificate = unavailable Then STOP 3. IF all clear Then OUTPUT “You are eligible for license” Otherwise OUTPUT “You are not eligible for license” 4. End	1. READ 2. SELECTION 3. READ 4. SELECTION 5. READ 6. SELECTION 7. READ 8. SELECTION 9. READ 10. SELECTION 11. SELECTION + OUTPUT 12. END	Either “eligible” or “ineligible”

Algorithm:

1. Start
2. Declare variables:
    a. Age

b. Eyesight_Test
c. Written_Test
d. Driving_Test
e. Medical_fitness_certificate

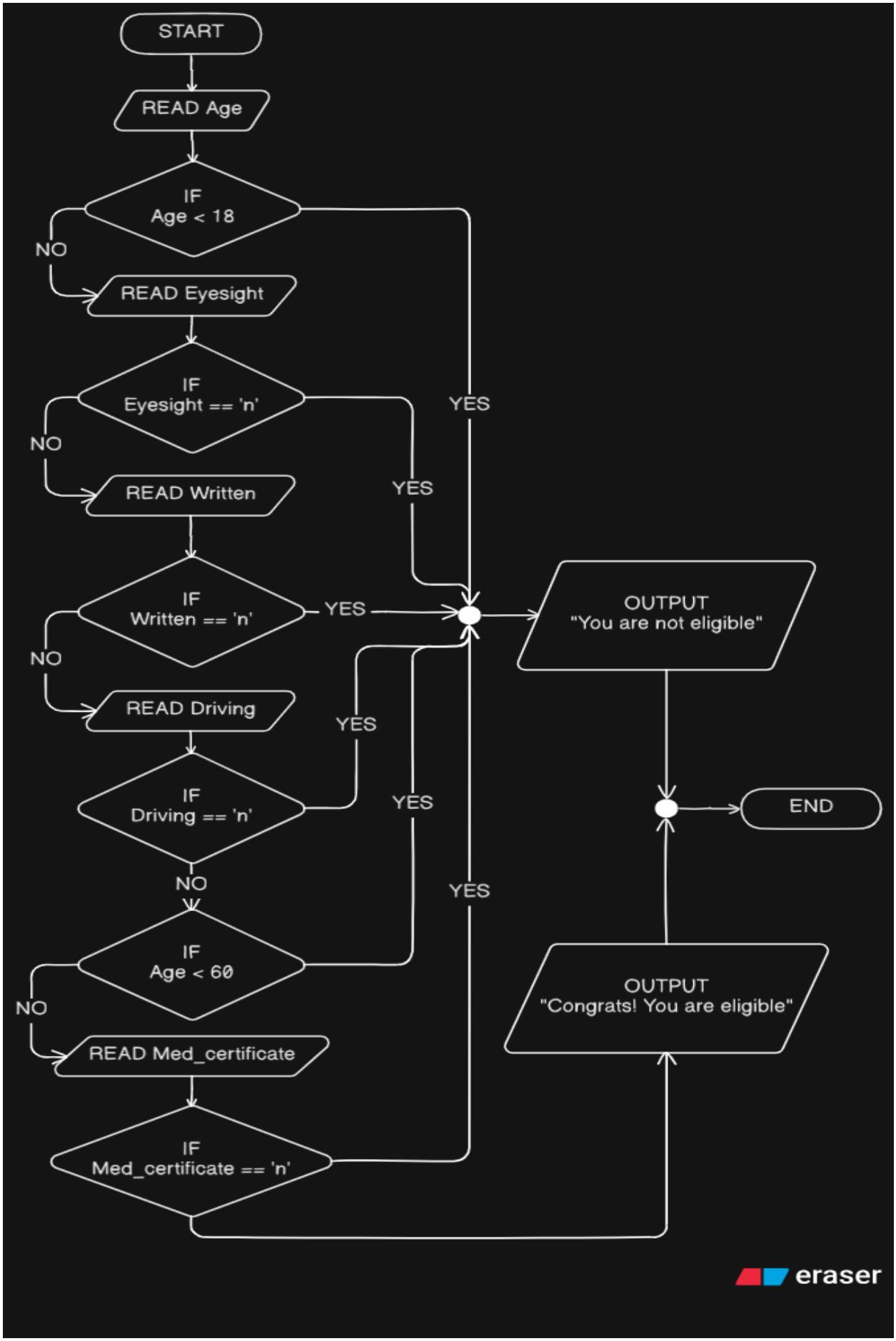
3. Read Age
4. Check If Age < 18 Then OUTPUT “You are not eligible” Otherwise continue...
5. Read Eyesight_Test result
6. Check If Eyesight_Test is failed (false) Then OUTPUT “You are not eligible” Otherwise continue...
7. Read Written_Test result
8. Check If Written_Test is failed (false) Then OUTPUT “You are not eligible” Otherwise continue...
1. Read Driving_Test result
2. Check If Driving_Test is failed (false) Then OUTPUT “You are not eligible” Otherwise continue...
1. Check If Age < 60 Then OUTPUT “You are not eligible” Otherwise continue...
9. Read Medical_fitness_certificate
10. Check IF Medical_fitness_certificate is unavailable (false) Then OUTPUT “You are not eligible” Otherwise continue...
11. If all conditions are true Then OUTPUT “You are eligible”
12. End

Pseudocode:

```
START

DECLARE

        Age (number)

        Eyesight (0/1)

        Written (0/1)

        Driving (0/1)

        Med_certificate (0/1)

READ Age

IF Age < 18:

        OUTPUT “You are not eligible”

        END

ENDIF
```

```
READ Eyesight
IF Eyesight == 0:
        OUTPUT “You are not eligible”
        END
ENDIF
READ Written
IF Written == 0:
        OUTPUT “You are not eligible”
        END
ENDIF
READ Driving
IF Driving == 0:
        OUTPUT “You are not eligible”
        END
ENDIF
IF Age < 60:
        OUTPUT “You are not eligible”
        END
ENDIF
READ Med_certificate
IF Med_certificate ==  0:
        OUTPUT “You are not eligible”
        END
ENDIF
OUTPUT “Congrats! You are eligible”
END
```

Flowchart: (next page) (link: https://app.eraser.io/workspace/v7VULijSGszJ5veWztsu?origin=share)


START
READ Age
IF
Age < 18
NO
YES
READ Eyesight
IF
Eyesight == 'n'
NO
YES
READ Written
IF
Written == 'n'
YES
NO
READ Driving
IF
Driving == 'n'
YES
NO
IF
Age < 60
YES
NO
READ Med_certificate
IF
Med_certificate == 'n'
YES
OUTPUT
"You are not eligible"
OUTPUT
"Congrats! You are eligible"
END
eraser

C Code:

```
#include <stdio.h>
#include <stdlib.h>

int main() {
  int age = 0, eyesight = 0, written = 0, driving = 0, med_certificate
= 0;

  printf("Enter your age: ");
  scanf("%d", &age);

  if (age < 18) {
    printf("You are not eligible.");
    exit(0);
  }

  printf("Did you pass the eyesight test? (0/1): ");
  scanf("%d", &eyesight);

  if (eyesight == 0) {
    printf("You are not eligible.");
    exit(0);
  }

  printf("Did you pass the written test? (0/1): ");
  scanf("%d", &written);

  if (written == 0) {
    printf("You are not eligible.");
    exit(0);
  }

  printf("Did you pass the driving test? (0/1): ");
  scanf("%d", &driving);

  if (eyesight == 0 || age < 60) {
    printf("You are not eligible.");
    exit(0);
  }

  printf("Do you have Medical Fitness Certificate? (0/1): ");
  scanf("%d", &med_certificate);

  if (eyesight == 0) {
    printf("You are not eligible.");
    exit(0);
```

```
  }

  printf("Congrats! You are eligible.");
  return 0;
}
```

Question 2:

PAC Chart:

Given Data	Required Result
Hand -> array of 5 integers	Whether full house or not
Processing required	**Solution Alternatives**
Read input Sort the array in ascending order Check if first three and last 2 elements are the same. Check if first 2 and last 3 elements are the same.	Use a faster sorting algorithm Use manual sorting Use iterations and loops

IPO Chart:

Input	Processing	Module Reference	Output
Hand: array of int[5]	1. Read 5 integers from user 2. Sort in ascending order 3. Check if Hand[0] == Hand[1] AND Hand[1] == Hand[2] AND Hand[3] == Hand[4] 4. Check if Hand[0] == Hand[1] AND Hand[2] == Hand[3] AND Hand[3] == Hand[4] 5. OUTPUT “It is/is not a full house” 6. End	1. READ 2. COMPUTE 3. SELECTION 4. SELECTION 5. OUTPUT 6. END	Whether the poker hand is full house or not.

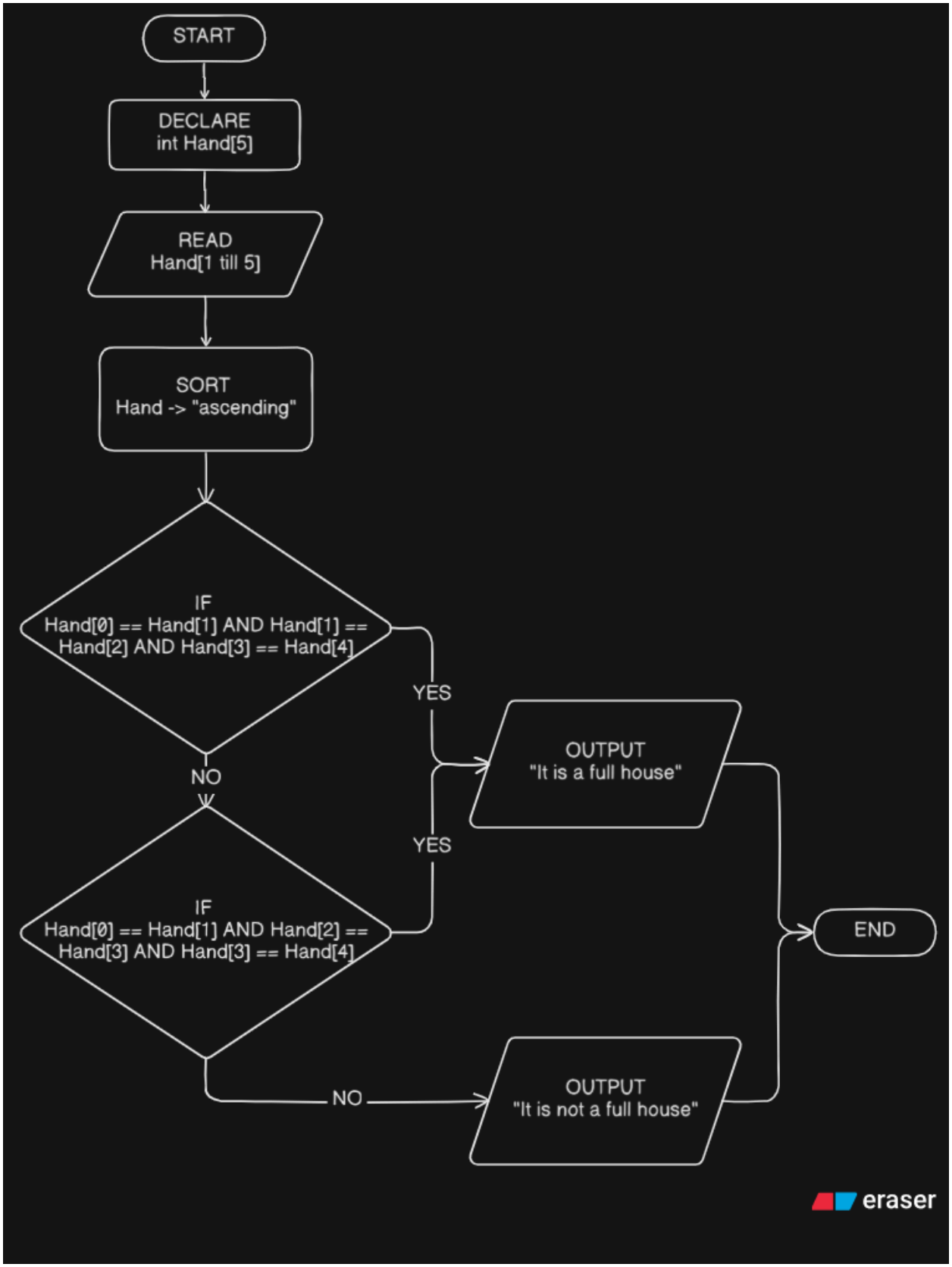
Algorithm:

1. Start.
2. Declare an array Hand of size 5.
3. Read 5 integers from user (1-13) and set into Hand array.
4. Sort the hand array in ascending order.
5. Check for a full house:
6. IF Hand[0] == Hand[1] AND Hand[1] == Hand[2] AND Hand[3] == Hand[4]
7. Then, print "This is a full house."
8. ELSE IF Hand[0] == Hand[1] AND Hand[2] == Hand[3] AND Hand[3] == Hand[4]
9. Then, print "This is a full house."
10. Otherwise:
11. Print "This is not a full house."
12. End.

Pseudocode:

```
START

DECLARE

    Hand: int[5]

READ

    Hand[0], Hand[1], Hand[2], Hand[3], Hand[4]

COMPUTE

    sort(Hand, “ascending”); //Use built-in function for simplicity

IF (Hand[0] == Hand[1] AND Hand[1] == Hand[2] AND Hand[3] == Hand[4]):

    OUTPUT “It is a full house.”

ELSE IF (Hand[0] == Hand[1] AND Hand[2] == Hand[3] AND Hand[3] == Hand[4]):

    OUTPUT “It is a full house.”

ELSE:

    OUTPUT “It is not a full house.”

ENDIF

END
```

Flowchart: (next page) (link: https://app.eraser.io/workspace/Atsk7UZVHTxWaeZLrY8E?origin=share)


START
DECLARE
int Hand[5]
READ
Hand[1 till 5]
SORT
Hand -> "ascending"
IF
Hand[0] == Hand[1] AND Hand[1] ==
Hand[2] AND Hand[3] == Hand[4]
YES
OUTPUT
"It is a full house"
NO
IF
Hand[0] == Hand[1] AND Hand[2] ==
Hand[3] AND Hand[3] == Hand[4]
YES
END
NO
OUTPUT
"It is not a full house"
eraser

C Code:

```
#include <stdio.h>

#include <stdlib.h>


int compare(const void *a, const void *b) { return (*(int *)a - *(int
*)b); }


int main() {

  int hand[5] = {0, 0, 0, 0, 0};

  int arr_size = sizeof(hand) / sizeof(int);


  printf("Enter 5 card numbers (1-13): ");

  scanf("%d %d %d %d %d", &hand[0], &hand[1], &hand[2], &hand[3],
&hand[4]);


  qsort(hand, arr_size, sizeof(int), compare);


  if (hand[0] == hand[1] && hand[1] == hand[2] && hand[3] == hand[4])
{

    printf("It is a full house.\n");

  } else if (hand[0] == hand[1] && hand[2] == hand[3] && hand[3] ==
hand[4]) {

    printf("It is a full house.\n");

  } else {

    printf("It is not a full house.\n");

  }


  return 0;

}
```

Question #5:

C Code:

```
#include <stdbool.h>
#include <stdio.h>

int main() {
  int n, count[10];
  printf("Enter a chain of numbers (0-9). Enter 2 digit number to
stop: ");
  while (true) {
    scanf("%d", &n);
    if (n < 0 || n > 9) {
      break;
    }
    count[n]++;
  }
  printf("Number\tNo. of Occurrences\n");
  for (int i = 0; i < 10; i++) {
    printf("%d\t%d\n", i, count[i]);
  }
}
```

Dry Run Table:

User Input (n)	if (n < 0 || n > 9)	Action	count Array Change
0	FALSE	count[0]++	count[0] = 1
0	FALSE	count[0]++	count[0] = 2
1	FALSE	count[1]++	count[1] = 1

1	FALSE	count[1]++	count[1] = 2
1	FALSE	count[1]++	count[1] = 3
1	FALSE	count[1]++	count[1] = 4
1	FALSE	count[1]++	count[1] = 5
1	FALSE	count[1]++	count[1] = 6
1	FALSE	count[1]++	count[1] = 7
1	FALSE	count[1]++	count[1] = 8
1	FALSE	count[1]++	count[1] = 9
1	FALSE	count[1]++	count[1] = 10
2	FALSE	count[2]++	count[2] = 1
2	FALSE	count[2]++	count[2] = 2
2	FALSE	count[2]++	count[2] = 3
3	FALSE	count[3]++	count[3] = 1
3	FALSE	count[3]++	count[3] = 2
3	FALSE	count[3]++	count[3] = 3
3	FALSE	count[3]++	count[3] = 4
4	FALSE	count[4]++	count[4] = 1
4	FALSE	count[4]++	count[4] = 2
4	FALSE	count[4]++	count[4] = 3
4	FALSE	count[4]++	count[4] = 4
4	FALSE	count[4]++	count[4] = 5
4	FALSE	count[4]++	count[4] = 6
4	FALSE	count[4]++	count[4] = 7
4	FALSE	count[4]++	count[4] = 8
5	FALSE	count[5]++	count[5] = 1
5	FALSE	count[5]++	count[5] = 2
5	FALSE	count[5]++	count[5] = 3

5	FALSE	count[5]++	count[5] = 4
5	FALSE	count[5]++	count[5] = 5
5	FALSE	count[5]++	count[5] = 6
5	FALSE	count[5]++	count[5] = 7
99	TRUE	BREAK	NO CHANGE

FINAL VALUE of *count = {2, 10, 3, 4, 8, 7, 0, 0, 0, 0};*

Question #6:

C Code (reference):

```
if(i<j){
	if(j<k){
		i = j;
}
else {
	j = k;
	if (j > k)
		j = i;
	else
		i = k;
}
printf("%d %d %d\n", i, j, k);
```

Dry Run Table:

Case	Initial Values (i, j, k)	i < j	j < k	Change	j > k	Change	Output (i, j, k)
a)	3, 5, 7	TRUE	TRUE	i=5	-	-	5 5 7
b)	3, 7, 5	TRUE	FALSE	-	-	-	3 7 5
c)	5, 3, 7	FALSE	-	j=7	FALSE	i=7	7 7 7
d)	5, 7, 3	TRUE	FALSE	-	-	-	5 7 3
e)	7, 3, 5	FALSE	-	j=5	FALSE	i=5	5 5 5
f)	7, 5, 3	FALSE	-	j=3	FALSE	i=3	3 3 3

Question #7:

Corrected Code:

```
#include <stdio.h>

int main() {
  int value;

  printf("Please enter a value in the range 1...7: ");
  scanf("%d", &value);

  if (value == 1) {
    printf("You entered Monday\n");
  } else if (value == 2) {
    printf("You entered Tuesday\n");
  } else if (value == 3) {
    printf("You entered Wednesday\n");
  } else if (value == 4) {
```

```
    printf("You entered Thursday\n");
  } else if (value == 5) {
    printf("You entered Friday\n");
  } else if (value == 6) {
    printf("You entered Saturday\n");
  } else if (value == 7) {
    printf("You entered Sunday\n");
  } else {
    printf("You entered a value out of range\n");
  }
}
```

Rewritten Code with Switch Case:

```
#include <stdio.h>

int main() {
  int value;

  printf("Please enter a value in the range 1...7: ");
  scanf("%d", &value);

  switch (value) {
  case 1:
    printf("You entered Monday\n");
    break;
  case 2:
    printf("You entered Tuesday\n");
```

```
    break;
  case 3:
    printf("You entered Wednesday\n");
    break;
  case 4:
    printf("You entered Thursday\n");
    break;
  case 5:
    printf("You entered Friday\n");
    break;
  case 6:
    printf("You entered Saturday\n");
    break;
  case 7:
    printf("You entered Sunday\n");
    break;
  default:
    printf("You entered a value out of range\n");
    break;
  }
}
```