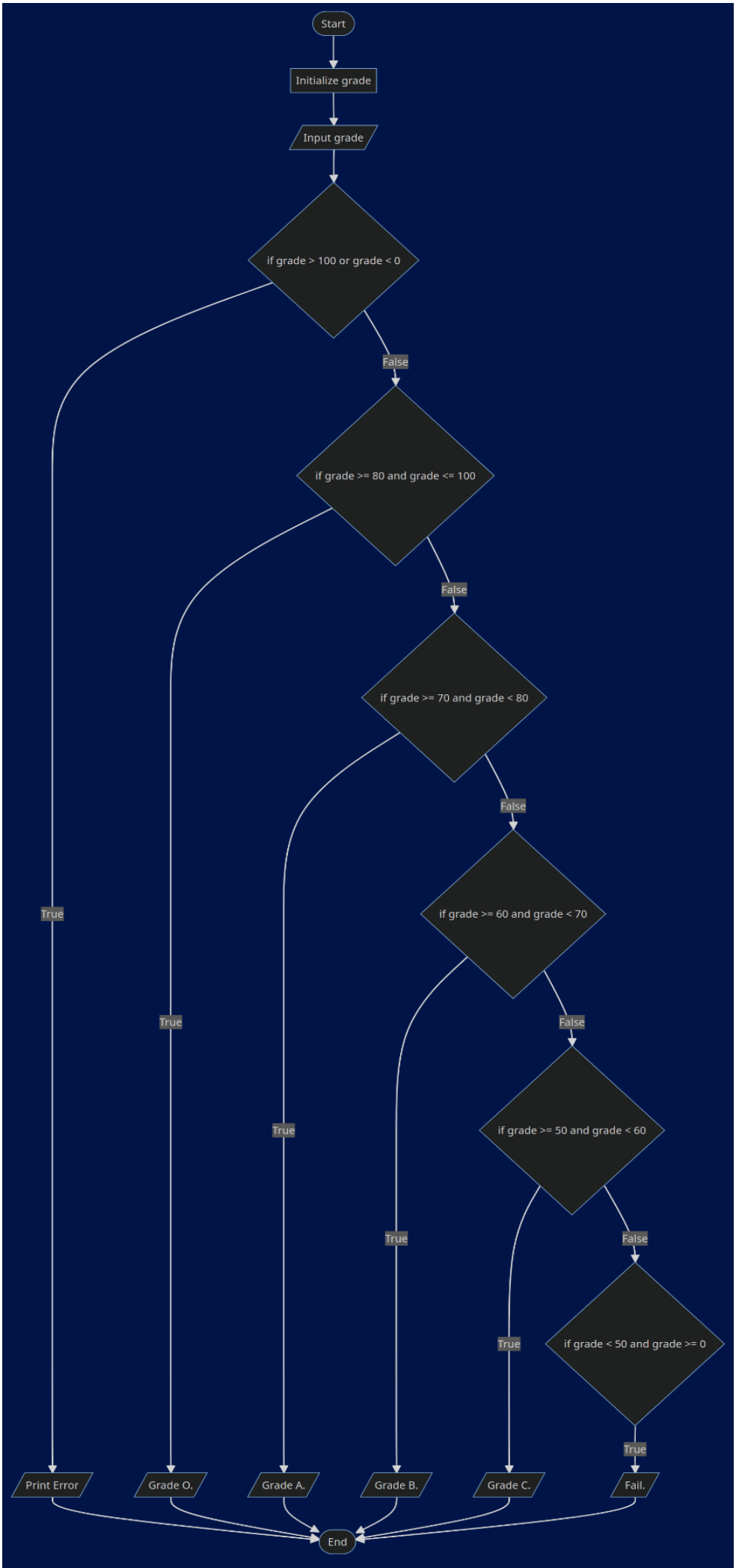
Name	Jhaveri Varun Nimitt
UID no.	2023800042
Experiment No.	2

AIM:	Apply various control structures to solve given problems.
Program 1	
PROBLEM STATEMENT :	Find the letter grade of a student marks using if else.
ALGORITHM:	1. Start 2. Declare an integer variable grade. 3. Input grade. 4. Read the value of grade from the user using scanf. 5. Check if grade is greater than 100 or less than 0: - If true, display Error. - Exit the program with code 0. 6. Check if grade is between 80 and 100: - If true, display "Grade O." and the program stops. - If false, continue to the next else if statement. 7. Check if grade is between 70 and 79: - If true, display "Grade A."and the program stops. - If false, continue to the next else if statement. 8. Check if grade is between 60 and 69: - If true, display "Grade B."and the program stops. - If false, continue to the next else if statement. 9. Check if grade is between 50 and 59: - If true, display "Grade C."and the program stops. - If false, continue to the next else if statement. 10. Check if grade is less than 50 and greater than or equal to 0: - If true, display "Fail." . 11. Stop

FLOWCHART:


Start
Initialize grade
Input grade
if grade > 100 or grade < 0
True
False
if grade >= 80 and grade <= 100
True
False
if grade >= 70 and grade < 80
True
False
if grade >= 60 and grade < 70
True
False
if grade >= 50 and grade < 60
True
False
if grade < 50 and grade >= 0
True
Print Error
Grade O.
Grade A.
Grade B.
Grade C.
Fail.
End

PROGRAM:

```c
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int grade;
    printf("Enter grade:");
    scanf("%d", &grade);
    if(grade>100 || grade<0)
    {
        printf("Error");
        exit(0);
    }
    if (grade >=80 && grade <=100)
    {
        printf("Grade O.");
    }
    else if (grade>= 70  && grade<80)
    {
        printf("Grade A.");
    }
    else if (grade >=60 && grade<70)
    {
        printf("Grade B");
    }
    else if (grade >= 50 && grade<60)
    {
        printf("Grade C.");
    }
    else if (grade <50 && grade>=03)
    {
        printf("Fail.");
    }
    return 0;
}
```

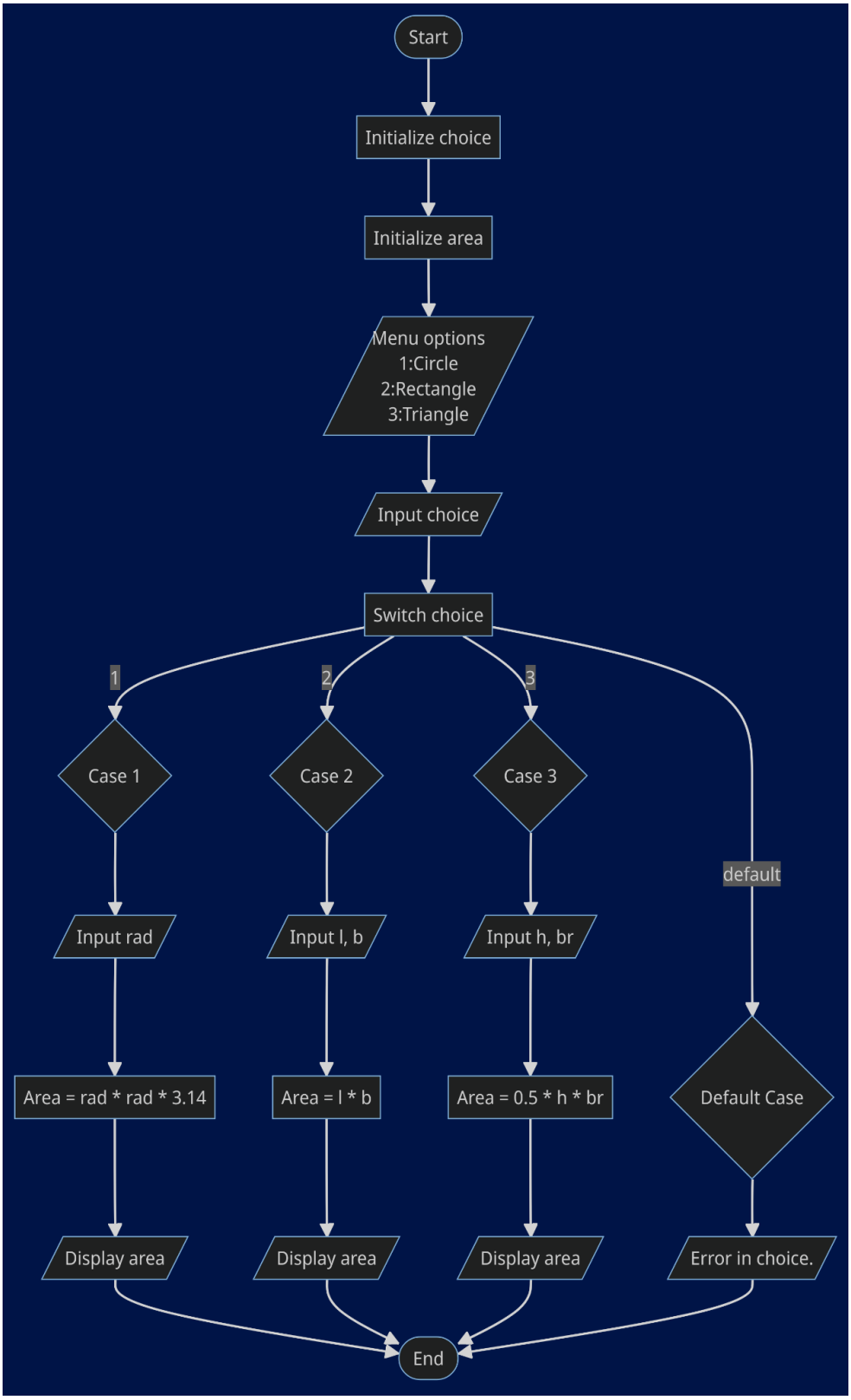
RESULT:

```
cyclops@cyclops:~$ cd Desktop/
cyclops@cyclops:~/Desktop$ cd PSIPL\ Semester\ 1/
cyclops@cyclops:~/Desktop/PSIPL Semester 1$ cd Experiment\ 2/
cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ gcc grade.c
cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter grade:92
Grade O.cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter grade:-3
Error.cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter grade:72
Grade A.cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter grade:62
Grade Bcyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter grade:52
Grade C.cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter grade:42
Fail.cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$
```

Program 2	
PROBLEM STATEMENT :	Write a program using switch case to find the area of different shapes.
ALGORITHM:	1. Start 2. Declare an integer variable choice and a double variable area. 3. Display the menu: - Enter 1 for circle. - Enter 2 for rectangle. - Enter 3 for triangle. - Enter choice: 4. Read the value of choice from the user using scanf. 5. Switch(choice) - Case 1: - Display Enter radius : - Declare a double variable rad. - Read the value of rad from the user using scanf. - Calculate the area as area = rad * rad * 3.14. - Display Area: area value.. - Break out of the switch statement. - Case 2: - Display Enter length, breadth : - Declare double variables l and b. - Read the values of l and b from the user using scanf. - Calculate the area as area = l * b. - Display Area: area value. - Break out of the switch statement.

	- Case 3: - Display Enter height, base : - Declare double variables h and br. - Read the values of h and br from the user using scanf. - Calculate the area as area = 0.5 * h * br. - Display Area: area value. - Break out of the switch statement. - Default case: - Display Error in choice. 6. Stop

FLOWCHART:



PROGRAM:

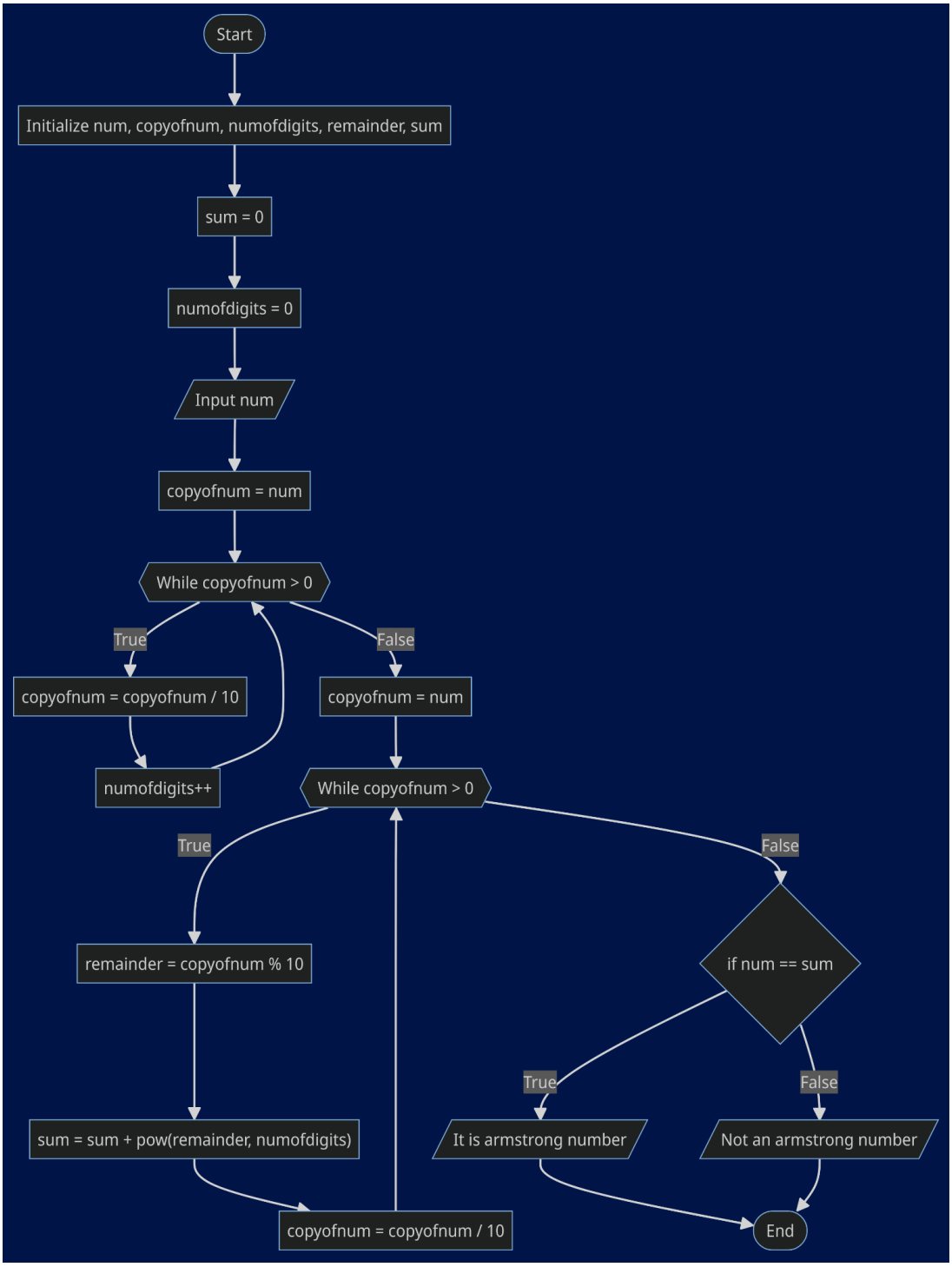
```c
#include <stdio.h>

int main()
{
    int choice;
    double area;
    printf("Enter 1 for circle.\n");
    printf("Enter 2 for rectangle.\n");
    printf("Enter 3 for triangle.\n");
```

```
    printf("Enter choice:");
    scanf("%d", &choice);
    switch (choice)
    {
        case 1 :
            printf("Enter radius :\n");
            double rad;
            scanf("%lf",&rad);
            area = rad*rad*3.14;
            printf("Area: %lf",area);
        break;

        case 2 :
            printf("Enter length, breadth :\n");
            double l,b;
            scanf("%lf %lf",&l, &b);
            area = l*b;
            printf("Area: %lf",area);
        break;

        case 3 :
            printf("Enter height, base :\n");
            double h,br;
            scanf("%lf %lf",&h, &br);
            area = 0.5*h*br;
            printf("Area: %lf",area);
        break;
        default :
            printf("Error in choice.");
   }
    return 0;
}
```

RESULT:

```
cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ gcc switch\ area\ of\ s
hape.c
cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter 1 for circle.
Enter 2 for rectangle.
Enter 3 for triangle.
Enter choice:1
Enter radius :2.31
Area: 16.755354cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter 1 for circle.
Enter 2 for rectangle.
Enter 3 for triangle.
Enter choice:2
Enter length, breadth :31.2 93.2
Area: 2907.840000cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.ou
t
Enter 1 for circle.
Enter 2 for rectangle.
Enter 3 for triangle.
Enter choice:3
Enter height, base :2.2 4.12
Area: 4.532000cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$
```

Program 3	
PROBLEM STATEMENT:	Write a program to find whether a number is armstrong or not using while loop.
ALGORITHM:	1. Start 2. Declare integer variables: num, copyofnum, numofdigits, remainder, and sum. 3. Initialize sum to 0 and numofdigits to 0 for no garbage values. 4. Ask for input and read the value of num from the user using scanf. 5. Copy the value of num to copyofnum. 6. while loop with the condition copyofnum > 0: - a. Divide copyofnum by 10. - b. Increment numofdigits by 1. 7. Copy the value of num to copyofnum. 8. while loop with the condition copyofnum > 0: - a. Calculate the remainder by taking copyofnum % 10 and store it in remainder. - b. Calculate sum as sum + pow(remainder, numofdigits). - c. Divide copyofnum by 10. 9. Check if num is equal to sum: - If true, display It is an Armstrong number. - If false, display Not an Armstrong number.

	10. Stop
FLOWCHART:	

PROGRAM:	`#include <stdio.h>`
	`#include <math.h>`
	
	`int main()`
	`{`
	`int num, copyofnum, numofdigits, remainder, sum;`
	`sum = 0;`
	`numofdigits = 0;`
	`printf("Enter Number : ");`
	`scanf("%d", & num);`
	`copyofnum = num;`

```c
    while (copyofnum > 0)
    {
        copyofnum = copyofnum / 10;
        numofdigits++;
    }
    copyofnum = num;
    while (copyofnum > 0)
    {
        remainder = copyofnum % 10;
        sum = sum + pow(remainder, numofdigits);
        copyofnum = copyofnum / 10;
    }
    if (num == sum)
    {
        printf("It is armstrong number");
    }
    else
    {
        printf("Not an armstrong number");
    }
    return 0;
}
```

RESULT:

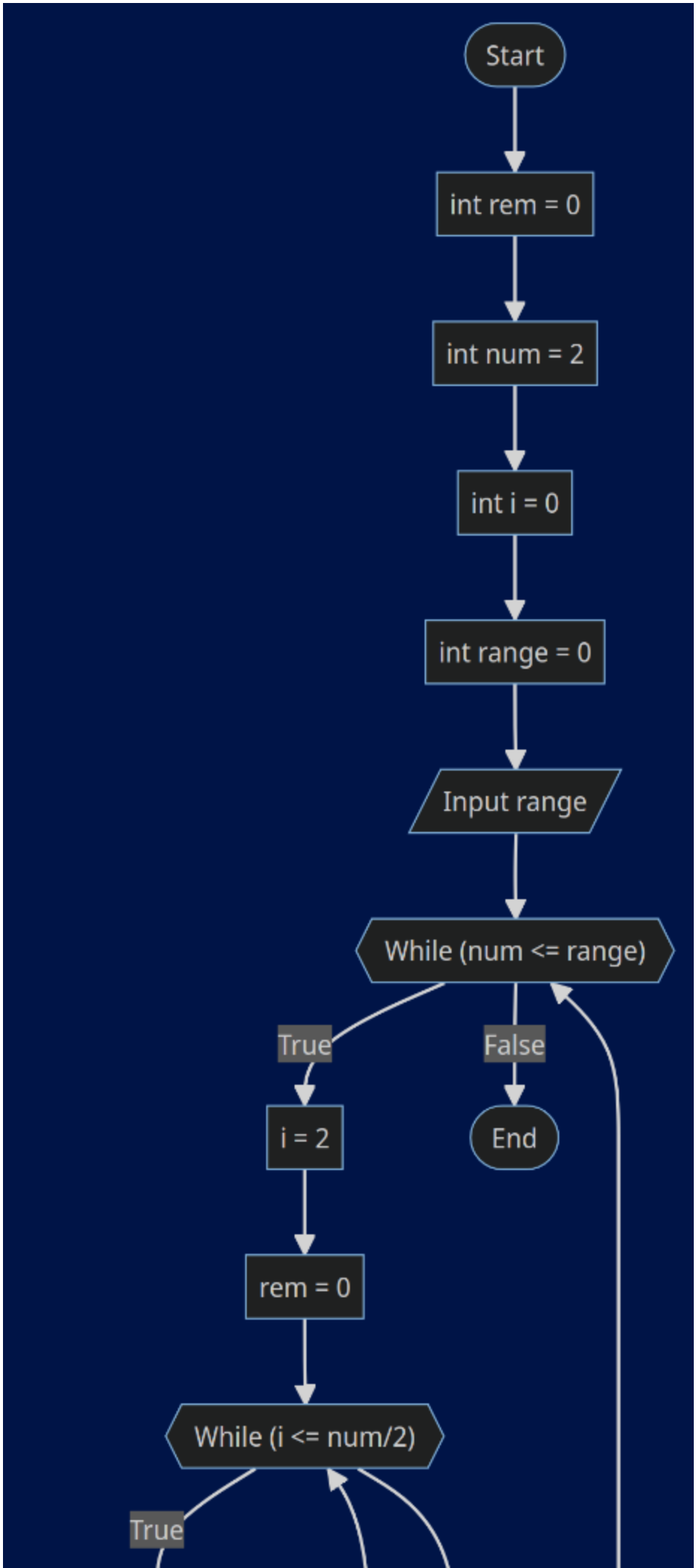
```
cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ gcc Armstrong.c -lm
cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter Number : 153
It is armstrong numbercyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ,
./a.out
Enter Number : 159
Not an armstrong numbercyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$
```

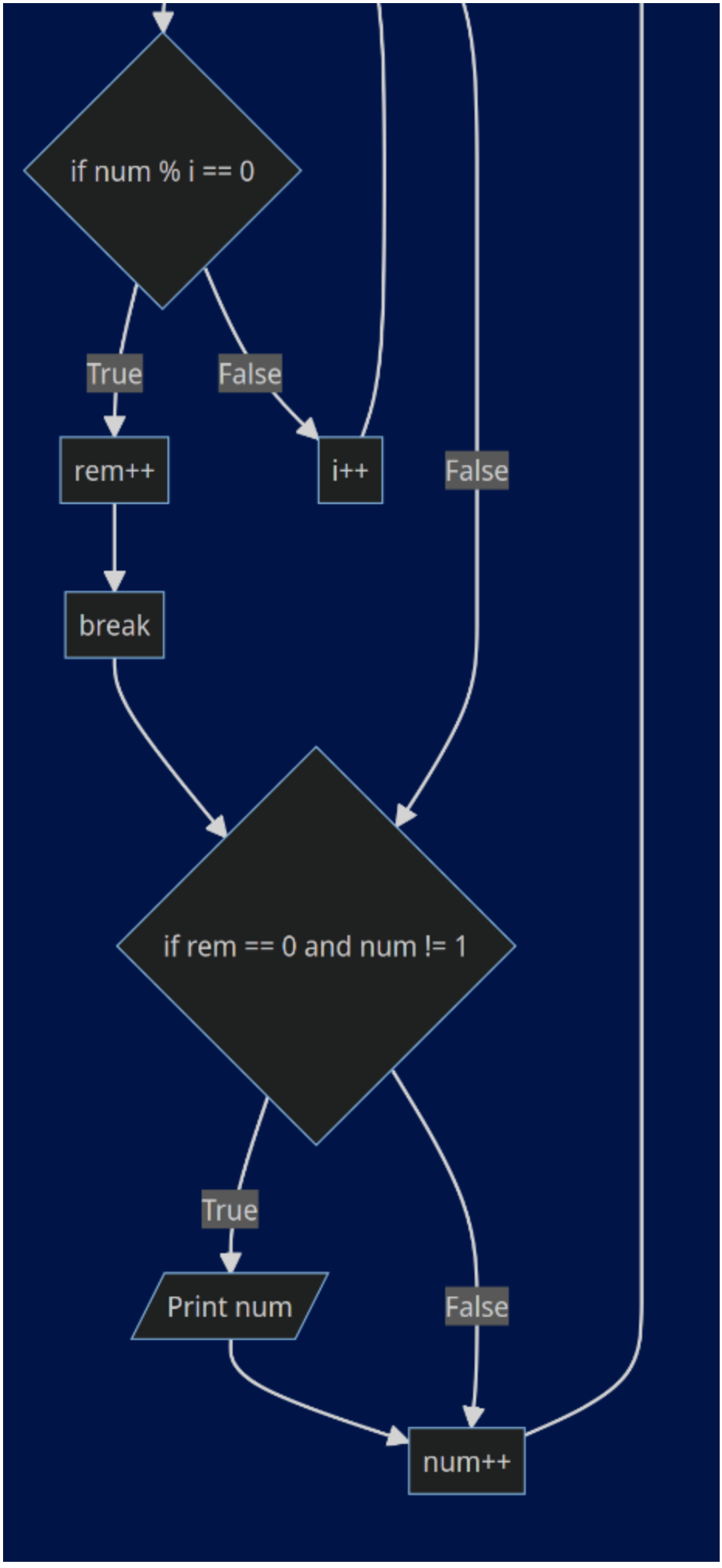
Program 4

PROBLEM STATEMENT:	Write a program to find the prime numbers within a range using for loop.
ALGORITHM:	1. Start 2. Declare integer variables: rem, num, i, and range. 3. Initialize rem to 0, num to 2, i to 0, and range to 0. 4. Input and read the value of range from the user using scanf. 5. Initialize a while loop with the condition num <= range: - a. Set i to 2. - b. Set rem to 0. - c. Initialize an inner while loop with the condition i <= num/2: - i. Check if num is divisible by i (aka num % i == 0):

	- If true, increment rem by 1. - Break out of the inner while loop. - ii. Increment i by 1. (End of inner while Loop) - d. Check if rem is equal to 0 and num is not equal to 1: - If true, display num. - e. Increment num by 1. 6. End of the outer while loop. 7. Stop

FLOWCHART:


Start
int rem = 0
int num = 2
int i = 0
int range = 0
Input range
While (num <= range)
True
False
i = 2
End
rem = 0
While (i <= num/2)
True


if num % i == 0
True
False
rem++
i++
False
break
if rem == 0 and num != 1
True
Print num
False
num++

PROGRAM:

```c
#include <stdio.h>

int main()
{
        int rem = 0;
        int num = 2;
         int i =0;
        int range = 0;
        printf("Enter range from 1 to:");
        scanf("%d", &range);
        while (num <= range)
        {
                i = 2;
                rem=0;
                while (i <= num/2)
                {
                        if (num % i == 0)
                        {
                                rem++;
                                break;
                        }
                        i++;
                 }
                 if (rem == 0 && num != 1)
                 {
                        printf("%d ", num);
                 }
                 num++;
        }
        return 0;
}
```

RESULT:

```
cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ gcc prime100.c
cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter range from 1 to:99
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 cyclops@c
yclops:~/Desktop/PSIPL Semester 1/Experiment 2$
```

Program 5

PROBLEM STATEMENT:

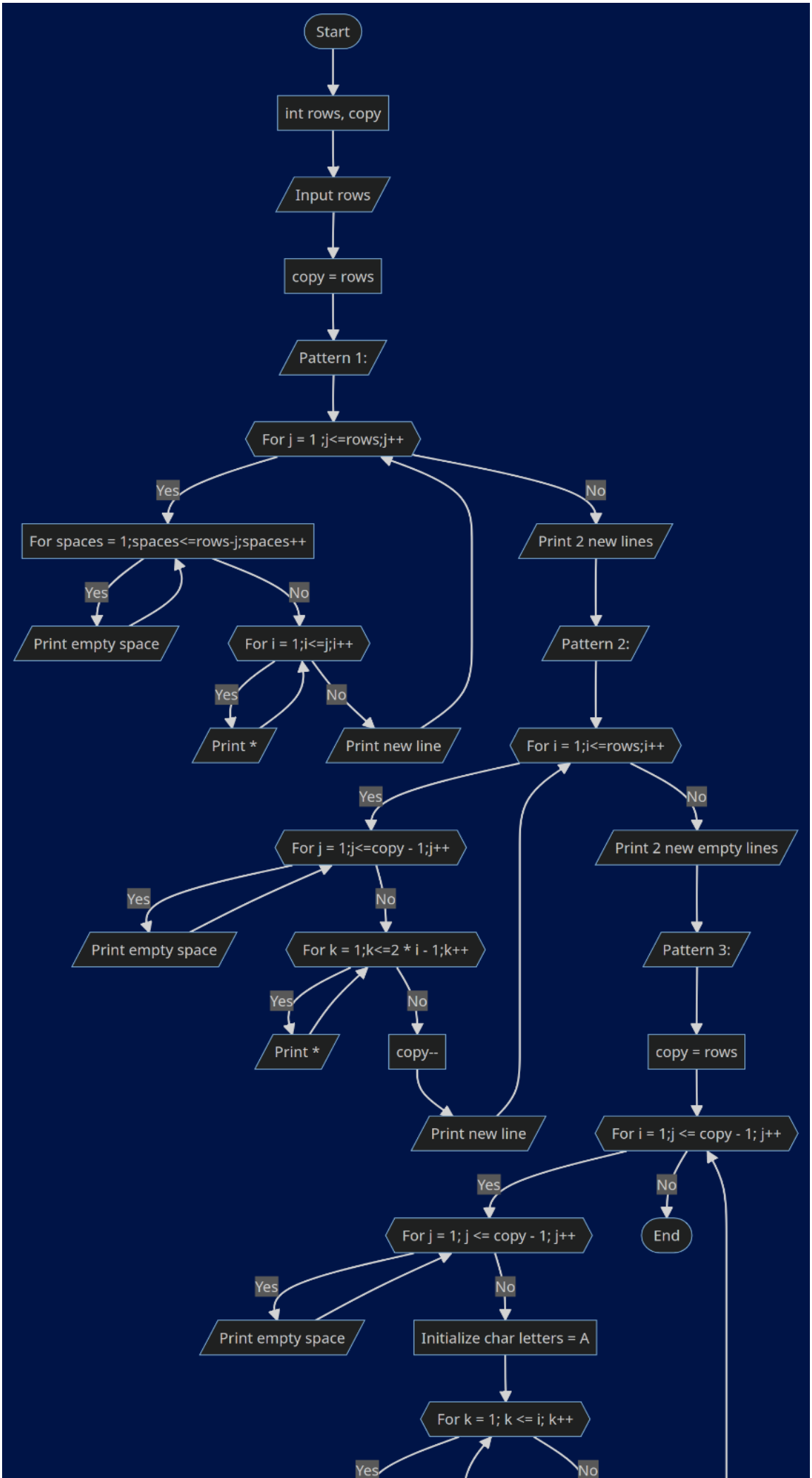
Write a program to print the following patterns(Take number of rows from the user):

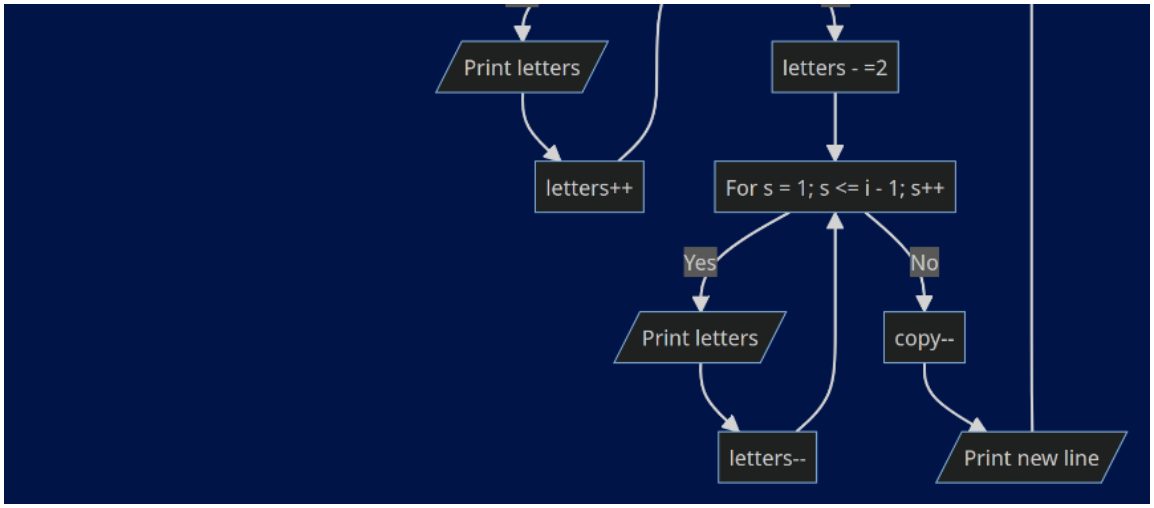
```
    *
   * *
  * * *
 * * * *
* * * * *
```

	* **A** **A B A** **A B C B A** **A B C D C B A** **A B C D E D C B A**
ALGORITHM:	1. Start 2. Declare integer variable rows. 3. Declare integer variable copy. 4. Read the value of rows from the user using scanf. 5. Copy the value of rows to copy. 6. Display: Pattern 1: 7. Loop j from 1 to rows (inclusive of final value) while incrementing j by 1 after each loop: - a. Loop spaces from 1 to rows - j (inclusive of final value) and print spaces while incrementing spaces by 1 after each loop. - b. Loop i from 1 to j (inclusive of final value) and print asterisks while incrementing i by 1 after each loop. - c. Print a newline. 8. Display two newline characters for separation. 9. Display: Pattern 2: 10. Loop i from 1 to rows(inclusive of final value) while incrementing i by 1 after each loop: - a. Loop j from 1 to copy - 1 (inclusive of final value) while incrementing j by 1 after each loop and print spaces. - b. Loop k from 1 to 2 * i - 1 (inclusive of final value) while incrementing k by 1 after each loop and print asterisks.

	- c. Decrement copy by 1. - d. Print a newline. 11. Display two newline characters for separation. 12. Display: Pattern 3: 13. Copy the value of rows to copy. 14. Loop i from 1 to rows (inclusive of final value) while incrementing i by 1 after each loop: - a. Loop j from 1 to copy - 1 (inclusive of final value) while incrementing j by 1 after each loop and print spaces. - b. Declare a character variable letters and initialize it to 'A'. - c. Loop k from 1 to i (inclusive of final value) while incrementing k by 1 after each loop - - Print the value of letters. - - Increment letters. - d. Decrement letters by 2. - e. Loop s from 1 to i - 1 (inclusive of final value) while incrementing s by 1 after each loop: - - Print the value of letters. - - Decrement letters. - f. Decrement copy by 1. - g. Print a newline. 15. Stop

FLOWCHART:


Start
int rows, copy
Input rows
copy = rows
Pattern 1:
For j = 1 ;j<=rows;j++
Yes
No
For spaces = 1;spaces<=rows-j;spaces++
Print 2 new lines
Yes
No
Print empty space
For i = 1;i<=j;i++
Pattern 2:
Yes
No
Print *
Print new line
For i = 1;i<=rows;i++
Yes
No
For j = 1;j<=copy - 1;j++
Print 2 new empty lines
Yes
No
Print empty space
For k = 1;k<=2 * i - 1;k++
Pattern 3:
Yes
No
Print *
copy--
copy = rows
Print new line
For i = 1;j <= copy - 1; j++
Yes
No
For j = 1; j <= copy - 1; j++
End
Yes
No
Print empty space
Initialize char letters = A
For k = 1; k <= i; k++
Yes
No



PROGRAM:

```c
#include <stdio.h>

int main()
{
	int rows, copy;
	printf("Enter the number of rows:");
	scanf("%d", &rows);
	copy = rows;
	printf("Pattern 1: \n");
	for (int j = 1; j <= rows; j++)
	{
		for (int spaces = 1;spaces<=rows-j;spaces++)
			printf(" ");
		for (int i = 1; i <= j; i++)
			printf("*");
		printf("\n");
	}
	printf("\n \n");
	printf("Pattern 2: \n");
	for (int i = 1; i <= rows; i++)
	{
		for (int j = 1; j <= copy - 1; j++)
		{
			printf(" ");
		}

		for (int k = 1; k <= 2 * i - 1; k++)
		{
			printf("*");
		}

		copy--;
		printf("\n");
	}
	printf("\n \n");
```

```
        printf("Pattern 3: \n");
        copy = rows;

        for (int i = 1; i <= rows; i++)
        {
                for (int j = 1; j <= copy - 1; j++)
                {
                        printf(" ");
                }

                char letters = 'A';
                for (int k = 1; k <= i; k++)
                {
                        printf("%c", letters);
                        letters++;
                }

                letters -= 2;
                for (int s = 1; s <= i - 1; s++)
                {
                        printf("%c", letters);
                        letters--;
                }

                copy--;
                printf("\n");
        }
        return 0;
}
```

RESULT:

```
cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ gcc Patterns.c
cyclops@cyclops:~/Desktop/PSIPL Semester 1/Experiment 2$ ./a.out
Enter the number of rows:6
Pattern 1:
     *
    **
   ***
  ****
 *****
******


Pattern 2:
     *
    ***
   *****
  *******
 *********
***********


Pattern 3:
     A
    ABA
   ABCBA
  ABCDCBA
 ABCDEDCBA
ABCDEFEDCBA
```

CONCLUSION	**I have understood the way to utilize loops and various control structures to solve problems.**