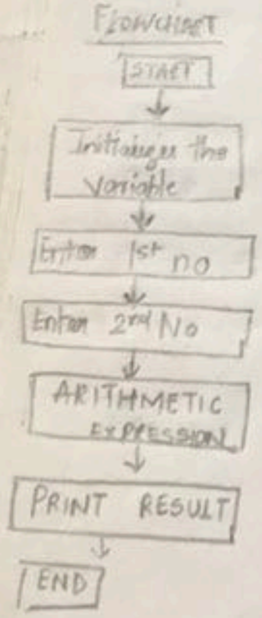
```
# Include<std io.h>
# Include<Conio.h>
void main()
{{
            int roll_no;
            char name[20], Mobile_no[10];
            float Percentage;
            clrscr();
            Printf("enter Student's name:\n");
            Scanf("%d", & name);
            Printf("enter Student's roll no:\n");
            scanf("%d", &rollno);
            printf("enter student's mobile no:\n");
            scanf("%s", &mobile_no");
            printf("enter student's percentage\n:");
            scanf("%f", &percentage");
            Printf("student's name: %s\n", name);
            printf("student's rollno: %d\n", roll_no);
            Printf("student's mobile_no: %s\n", mobile_no);
            Printf("student's percentage: %f\n", Percentage);
            getch();
}
```



```
output: Enter student's name: AKShatha
        Enter roll no: 1864
        Enter mobile no: 996788960
        Enter Student percentage: 82.06%
```

Programming with "C"

Practical-1

Q Write a C program to understand the basic datatype & Input/Output i.e name, Rollno, Phone no, percentage

Theory

Step1 - Declare a variable name, roll no as integer, also declare name, mobile no as characters & percentage as float.

Step2 - Use print f function to print question for user in order to give input

Step3 - Use scanf function to read users input & store in its allocated memory

Step4 - Again use print f function to display the output

Conclusion - The given program gives user an idea about how built-in datatypes work in & also about how user can give i/p & display output.

Nmeel

Practical No-2

Q1 Write a program on operators & expressions

Theory
step1 - Declare a variable name for first & second number of an integer

Step 2 - Now use scanf function to receive input from user

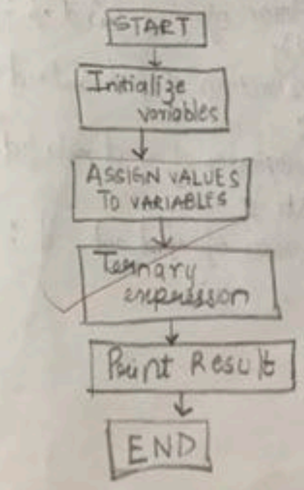
Step 3 - Now to add two numbers given by the user, use expression

Step 4 - x and y ~~Now to sub~~ use printf function to display the output

A) This program is for Addition of two numbers

```
# include <stdio.h>
# include <conio.h>
void main ();
{
        int x, y, z;
        clrscr();
        printf ("enter two numbers\n");
        scanf ("%d;%d", &x . &y);
        Z= X+ y;
        printf ("Sum of the two number= % d\n", z);
        getch ();

}
```

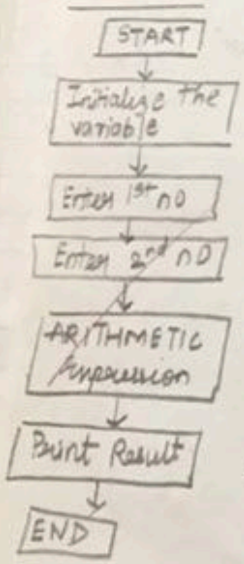
conclusion ~~output~~ Sum of the two numbers z = 8 + 9
= 15



B] The program is for calculator

```
#dynamic calculator
# include <stdio.h>
# include <conio.h>
void main()
{
      int num1, num2;
      float add, sub, mult, div;
      clscr();
      printf("enter first number:\n");
      scanf("%d", &num1);
      printf("enter second number:\n");
      scanf("%d", &num2);
      add = num1 + num2;
      sub = num1 - num2;
      multi = num1 x num2;
      div = num1 ÷ num 2;
      printf("Addition of %d and %:\n", num
          add);
      printf("Subraction of %d and .d:\n", num
          sub);
      printf("Mutiplication of %d and %d:\n", n
          multi);
      printf("Division of %d and %d : \n", nu
          div);
      getch();
}
```

FLOWCHART



B] Step1 - Declare a variable name for first & second number as integer

Step 2 - Now, use scanf function to recieve input from user

Step3 - Now to add two numbers given by the user, use the expression num1 + num 2

Step 4 - Now to subtract two numbers given by use expression num1 - num 2

Step5 - Again use expression num1 * num if user wishes to multiply the two input

Step6 - Use expression num1|num2 if user wishes to divide the two inputs

Step7 - Now use printf function to display output

Conclusion - These programs help us in having better understanding about operators & expressions

Practical No-3

Q Write a program on decision statements

For (A) Theory

Step1 - Declare a variable as integer and assign its values i.e 20

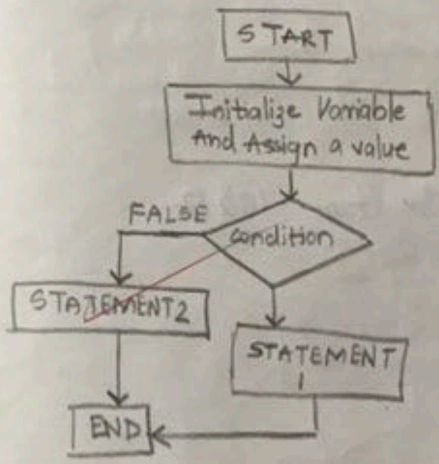
Step 2 - Now to compare whethere 20 is greater than 15 use if statement.

Step3 - If the conditioned is false, skip the if statement & print I am not in if

A] This program is for IF statement

```
#include <stdio.h>
#include <conio.h>
void main( )
{
        int i =10;
        clrscr();
        if(i > 15)
{
        printf("10 is less than 15\n");
}
        printf("I am no in f\n");
        getch();
}
```

Output - I am not in if

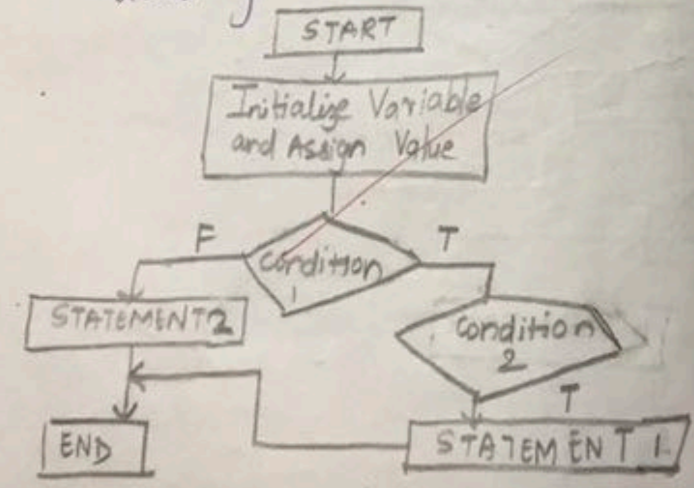


B] This Program is for IF ELSE statement

```
#include<stdio.h>
#include<conio.h>
void main ()
{
        int i=20;
        clrscr();
        if(i<15)
{
    printf("i is smaller than 15\n");
}
    else
{
    printf("i is greater than15\n");
}
    getch();
}
```

Output -

20 is greater than 10&12



This is (B) Theory

Step1 - Declare a variable as integer & assign its value i.e 20.

Step 2 - Now to compare the given value if its greater or not use if else conditional statement.

Step3 - If condition is true then print 20 is less than 15 or if condition is false then print 20 is greater than15.

This is © Theory

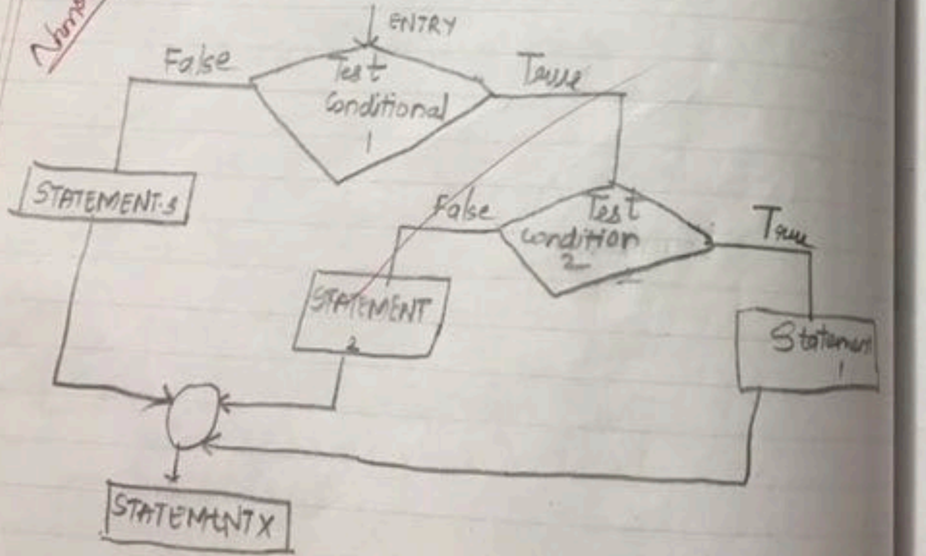
Step1 - Declare the variable as integer and assign value i.e 20

Step 2 - Now use nested if given no is greater or not

Step3 - If first condition is true then go to second condition if the condition is also true then print that 20 is greater than 15 & 12.

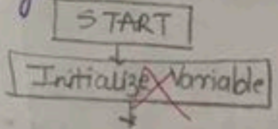
Conclusion : These programs help us to understand the working of if , if else & nested if conditional statement.

Nimode



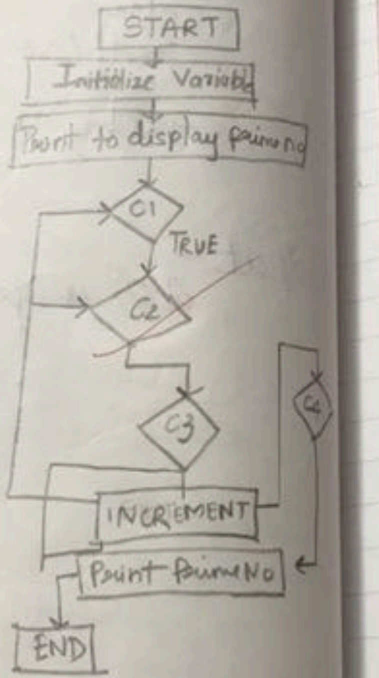
```
// This program is for NESTED IF
#include <stdio.h>
#include <conio.h>
void main()
{
        int = 10;
        clrscr();
        if(1==10);
{
        if (i<15)
        printf("i is smaller than 15\n");
        if (i<12).
        printf("i is smaller than 15\n");
        else
        printf("i is greater than 15\n");
}
        getch();
}
```

Output - 20 is greater than 10 & 12



PROGRAM

```
#include <stdio.h>
#include <conio.h>
void main ()
{
   int n, i, a;
   clrscr();
   printf("The prime no's are :");
   for (i=2; i<=20; i++)
   {      a=0;
          for (n=2; n < (i+1) /2; n++);
          { if (i % n==0)
              {
                a++;
              }
          }
          if (a==0)
          { printf("%d \n", i);
          }
   }
}
getch ();

}
```



Output - The prime no's are : 2
3
5
7
11
13

Practical - 4

Aim - To display prime number using for loop

Algorithm

Step1 - Initialize three variables out of which two are loop variables & count variables

Step2 - ~~Put~~ ~~anothere~~ Initialize a for loop from 2 to 50 set the count variable to zero

Step3 - Put anothere loop within a loop in step2 that goes from 2 to the first loop variable % 2.

Step4 - Use the if conditional statement to check whethere (1st loop variable % 2nd variable ==0) if true increment count variable ==0) if true increment count variab

Step5 - Come out of the second loop & check whethere the count variable is 0 if true print the number.

Step6 - Terminate the program

Conclusion - Prime number was displayed using for loop.

Q Write a C program to execute fibonacci series

Step1 - Start turbo c

Step2 - Declare variable n1, n2, n3, i & number

Step3 - Initialize the variables n1=0, n2=0 & number

Step4 - Enter the no of terms of fibonacci series to be printed

Step5 - Print 1st two nos of series as n1=0 & n2=1

Step6 - Use the for loop as per following step
n3 = n1+n2; n1=n2 ; n2=n3

Step7 - Increase the value of i element each time by 1

Step8 - Print the value of the n3

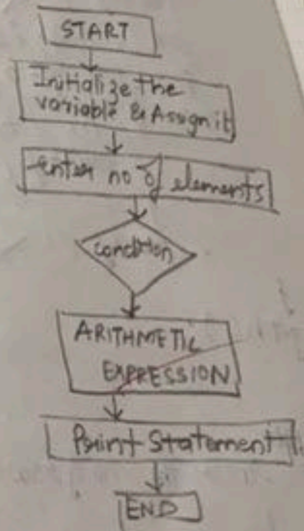
Conclusion - Thus we have successfully executed fibonacci series on turbo C.

program

```
#include <stdio.h>
#include <conio.h>
void main()
{ int n1=0,n2=1,n3,i,number;
printf ("enter no of elements\n");
scanf ("%d", &number);
printf ("%d %d", n1,n2);
for (i=2;i<= number;++i)
{
n3 =n1+n2;
printf ("%d", n3);
n1=n3;
n2=n3;
}
getch();
}
```

output

```
Enter no of elements.
6
112358
```



Practical-5

A] write a C program to find largest number among array

Algo.

Step1 - Start the C applications

Step2 - Declare the variable i & int array a[10]

Step3 - Enter the for loop at i=10. i<10 & use the value of a[i] till i<10. exit the for loop

Step4 - Enter the for loop at i=0, i<10 use if condition statement.

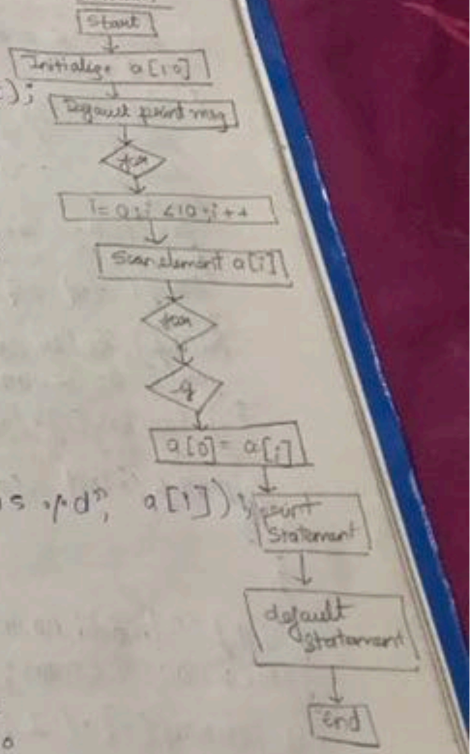
```
#include <conio.h>
# include <stdio.h>
void main()
{   int a [10], i;
    clrscr();
    printf("Enter the element of the int);
    for (i = 0; i<10; i++);
    {
     scanf("%d", & a[i]);
    }
    for (i= 0); i<10; i++)
    {
      if (a[0] < a[i]]
      {
        a[0] = a[i];
      }

    }
    printf("The layout number is %d", a[1]));
    getch();

}
```

output

```
12      2
        12
23      35      1
                2 2
2       55      100
```

The largest no is 100

FLOWCHART



CODE

```
#include<stdio.h>
# include <conio.h>
void main()
{ int array[100], i, num;
  printf("Enter the size of array \n");
  scanf("%d", &num);
printf("Enter the elements in array \n");
  for(i=0; i<num ; i++)
{scanf("%d", &array[i]);
}
printf("Enter even no in array");
 for(i=0; i<num ; i++)
{ if (arrary[i]%2 = = 0)
{ printf("%d", array [i] )
 }
}
}
printf("\n odd no in the array are");
for(i=0; i<num; i++)
{ if (array[i]%2 != 0)
{ printf("%d", array[i])
}
}
getch();
}
```

output

```
Enter the size of array
4
Enter the elements of array
5
3
8
1
Even no → ~~1 3 5~~ 8
Odd → 1 3 5
```

Q) Write a C-program to print the no of even and odd in the array.

Algo

Step1 - Create an array Takes its size from the user & define its element using loop.

Step2 - Display the size of the array

Step3 - Display the element of array entered by user.

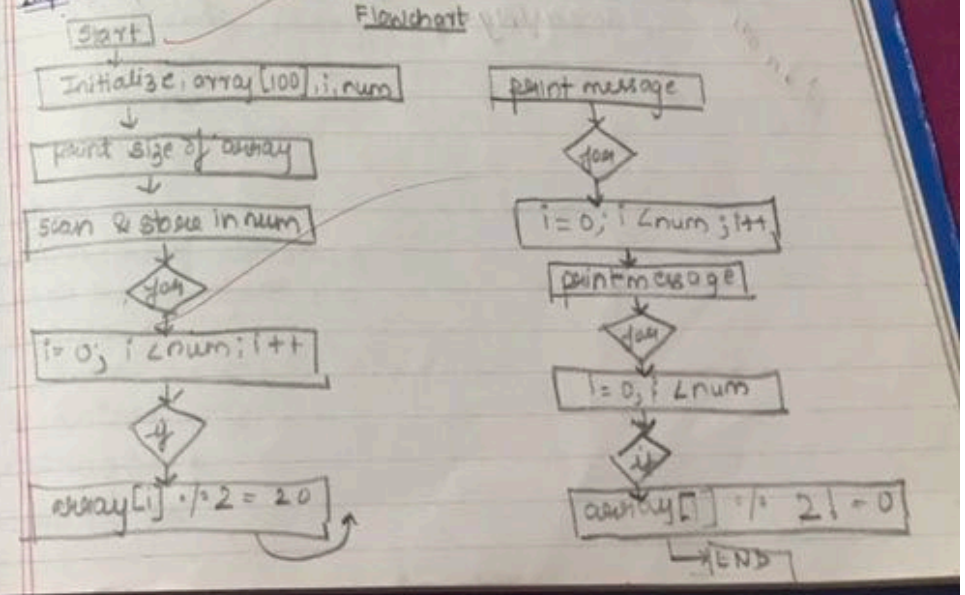
Step4 - Take the initializ initiator in a for loop using which all the elements of the array exist

Step5 - Display the even no from the array using for ~~for~~ loop : if (array[i]%2 == 0)
Display the even no from the given array.

Step6 - Display the odd no if (array[i]%2 ≠ 0]
then display odd no from the array

Step7 - Terminate the turbo C

Flowchart



c] **Aim**: WAP to find the sum or avg of element in an array.

* ALGOR-

Step1 - Start Turbo C application

Step2 - Declare int variable, n, i Initialize num [100], sum-0.0; avg.

Step3 - Using for loop at i=0; i<n; i++ give print message and increment by 1.

Step4 - Declare sum variable and store it by adding num [i].

Step5 - Average is sum divided by n

Step6 - give print statement for avg & sum

Step7 - Terminate

CONCLUSION. Thus we haveexecuted the program successfully.

Nimode

CODE

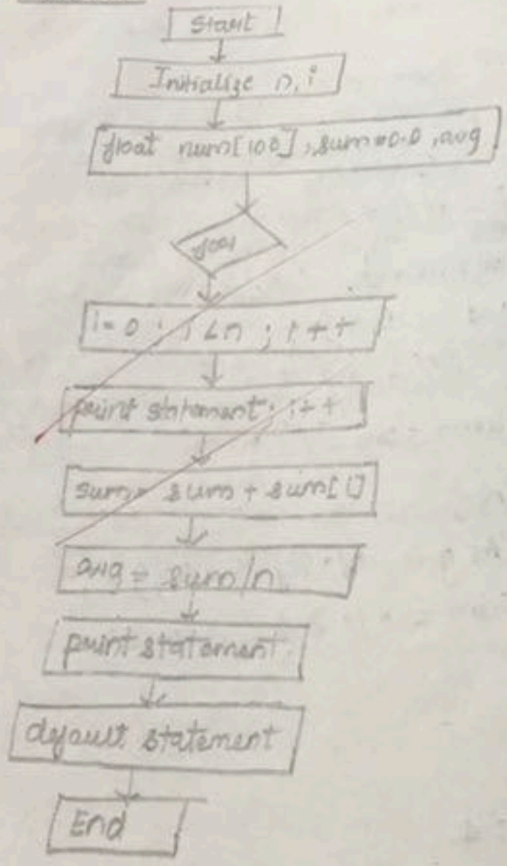
```
# include <stdio.h>
# include <conio.h>
void main()
{     int n, i;
      float num [100], sum=0.0, avg;
      clrscr ()
      printf ("Enter the noof elements");
      scanf ("%d", &n);
      for (i=0; i<n; i++)
      { printf("Enter no %d, i+1);
        scanf ("%d", &n);
        for(i=0; i<n; i++)
        { sum = sum+sum [i];
        }
        avg= sum/n;
        printf ("Avg = %.2f", avg);
        printf ("Sum = %.2f", sum);
        getch ();
      }
```

Output

```
Enter the no of elements: 4
enter the no again : 3
1. Enter no : 4
2. Enterno : 5
3. Enter no : 6
Average = 5
Sum is 15
```

040

Flowchart



PRACTICAL-6

Aim - Factorial of a number using recursion.

Algo -

S.1 - Start Turbo C application.

S.2 - Declare integer factorial(int, n)

S.3 - Using if statement, check if n>=1 & return

```
#include <stdio.h>
#include <conio.h>
int factorial(int n)
{       if (n>=1)
             return n * factorial(n-1);
        else
             return 1;
}
void main()
{       int n, a;
        clrscr();
        printf("Enter a positive integer");
        scanf("%d", &n);
        a = factorial(n);
        printf("\n Factorial of %d is %d", n, a);
        getch();
   }
```

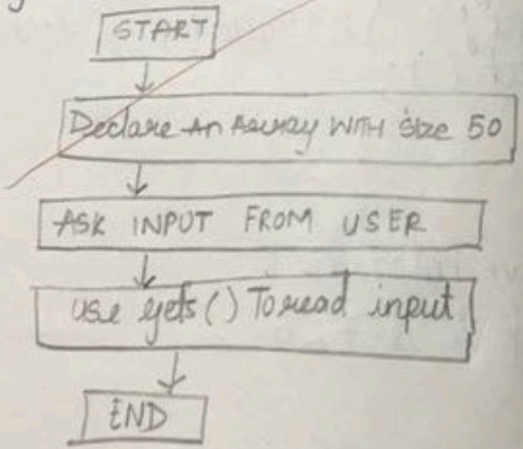
OUTPUT

```
Enter a +ve integer : 6
Factorial of 6 is 720
```

```
#include <string.h>
#include <conio.h>
#include <stdio.h>
 void main ()
{
   char name [50];
   printf ("\n please enter your name :");
   gets (name);
   printf ("%s", name);
   getch();
}
```

OUTPUT

please enter your name: Akshatha Swamy



Aim

2.) ~~Who~~ WAP to which shows the use of gets(

function.

Algo

S.1 - open Turbo C application.

S2 - Initialize character name [50].

S.3 : Use print statement to display your n

S.4 - use gets (name) to display your name.

S.5 - Use print statement to finally display u

nameo on o/p sceen.

S.6 - Terminate the program

Conclusion - Thus we have executed the program

Aim

Q3] WAP to which shows the use of puts function.

Algo

S1 - Open turbo C application

S.2 - Initialize character name [60]

S.3 - Use print statement to enter your name.

S.4 - Declare gets(name

S.5 - Use print statement to print the entered name.

S.6 - Use puts(name) to finally print entered name onto the output screen.

Conclusion :

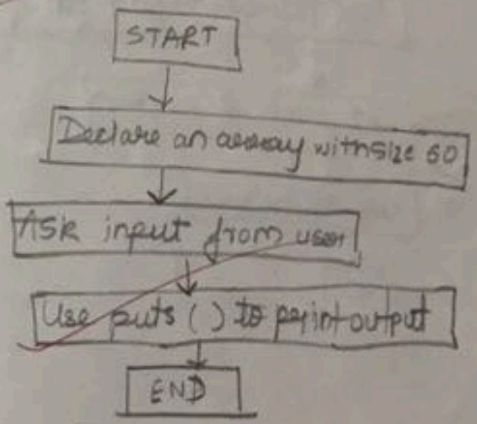
Thus we have successfully executed the program.

CODE :-

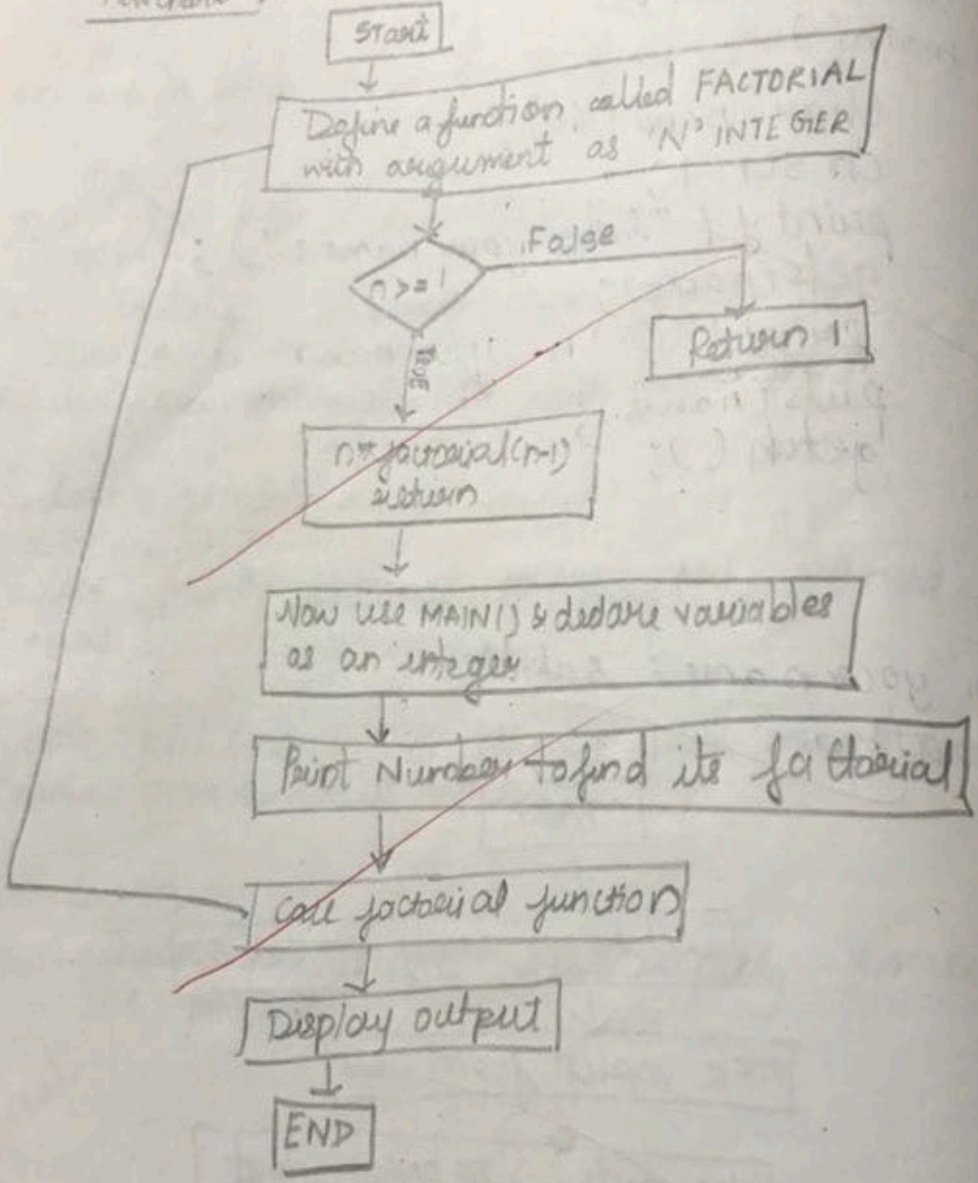
```
#include <string.h>
#include <conio.h>
#include <stdio.h>
void main()
{
        char name[60];
        clrscr();
        printf ("Enter your name :");
        gets(name);
        printf ("\n your name is ");
        puts(name);
        getch();
}
```

OUTPUT

```
Enter your name : sakshi
Akshatha.
```



Flow chart :-


Start
Define a function called FACTORIAL with argument as 'N' INTEGER
n>=1
False
Return 1
TRUE
n*factorial(n-1) return
Now use MAIN() & declare variables as an integer
Print Number to find its factorial
call factorial function
Display output
END

PRACTICAL-7

Aim - swapping of array

Algo -

Step1 - Start the turbo C

Step2 - Declare a function prototype with two integer pointer as argument before entering main ( )

Step3 - Declare 2 variable and accept the value from the user print the respective value using printf ( )

Step4 - for the address of the application as argument for the function

Step5 - Print the basic algorithm in the function definition but instead of normal variable user

Output

Enter the number to be swapped : 30 and 62
The no before swapping : 30 and the no after swapping : 6

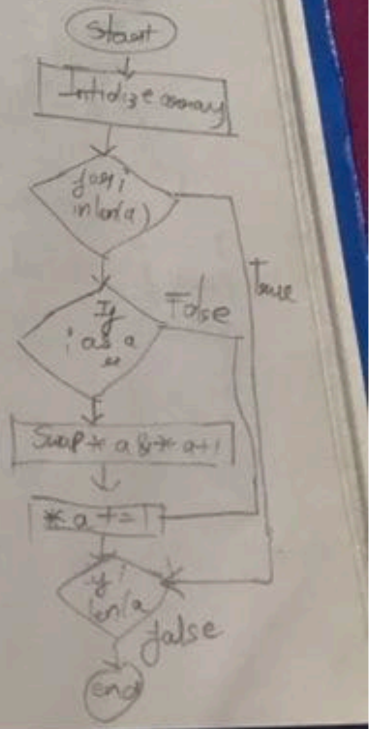
Conclusion

We have successfully executed the program

#code

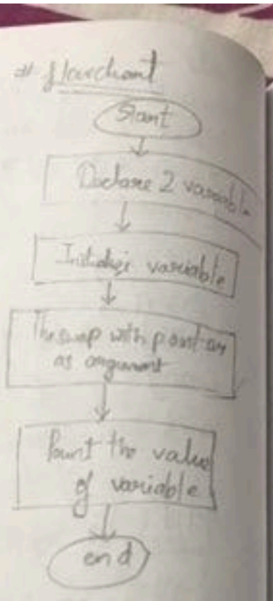
```
#include <conio.h>
#include <stdio.h>
void swamp (int *m, int *n);
void main( )
{ int x,y;
  clrscr();
  printf("Enter the two number to be swapped : ");
  scanf(" %d %d, &n, &m)
  printf ("The value of before swapping are %d and
         %d respectively x,y);
  getch( ):
}
void swa~~m~~p (int *m, int *n)
{ temp= *m;
  *m= * n;
  *n -temp;
}
```

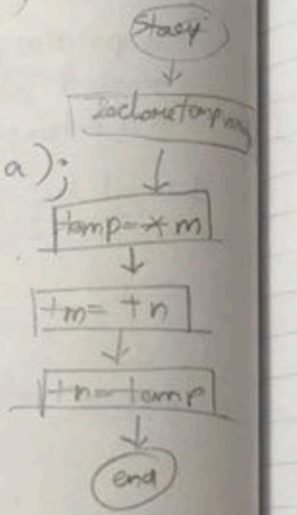
#flowchart



```
void sort (int n, int *p)
#include <stdio.h>
#include <conio.h>
void main ( )
{int a[10] ; i ; temp;
Clrscr();
for(i = 0; i<10 ; i++)
{
for (j=0 ; j <10 ; j++)
{ if (*a > *a +1)
{
 temp = * a+1;
 *a+1 = *a;
 *a = temp;              Swap(+m,+n)
}}
 }
}
Print f ("%d is a sorted array", a);
getch();
}
}
```

Flowchart


Start
Declare 2 variable
Initialize variable
The swap with pointer as argument
Print the value of variable
end



ii) Sorting of array using pointer

~~Algo~~

Step1 - Initialize an integer array and temp variable.

Step2 - Run a nested loop on i=0 len(a) and or i=0 to len(a)-i

Step3 - if *a > *a+1, swap the two values using basic swapping logic

Step4 - Print the swapped array

Step5 - Terminate the Program.

Output

Insert element into the array

1
9
5
8
2
3

{ 1, 2, 3, 5, 8, 9} is the sorted array

Conclusion - We have successfully executed the program

iii) Aim - one dimensional array representation using pointer

Algo

Step1 - Open turbo C7 program

Step2 - Initialize an integer array and a variable

Step3 - Run a while loop with i=0 + to len(a)

Step4 - Print the data of the array and then use pointer to print array location

Step5 - Terminate the program.

Output : The address of [0] = 65516
The value of a[0] - 7
The address of [2] = 65520
The value of a[2] = 4
The address of [4] = 65524
The value of a[4] - 9

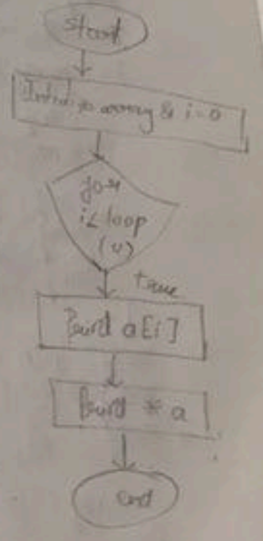
Conclusion - We have successfully executed the program

Nimodi

code

```
#include <stdio.h>
# include <conio.h>
void main ()
{int a [5] = {7,8,4,5,9};
 int * ptr;
 int i - 0;
 ptr = & a[0]
 while (*ptr! = '\0');
{print f (" \n The address of a[%d] - %d",i,ptr);
print f (" \n The address of a[%d] - %d",i,ptr);
 ptr ++;
 i ++;
}
getch ( );
}
```

#flowchart



Practical-8

A] Create a simple structure named student that holds the following variable
i) id ii) CGPA iii) Name

Algo

Step1 - Open turbo C++ program. As per question Use struct student.

Step2 - Initialize int for id, float ~~fo~~ for CGPA & char for Name

Step3 - Declare 2 variable from the user (size, i)
Thereby followed by printf statement (used in size).

Step4 - Use the for loop so that more than 1 records can be stored.

Step5 - Now use printf function so that the user can give input for the initialized id, cgpa & name.

Step6 - Use 2 printf statement after for loop, one for ~~enter~~ the student's data & the other to represent data in a tabular form.

Step7 - Again enter for loop statement so as to print more than 1 records. Now terminate the program.

OUTPUT -

```
How many records you want to insert : 2
0. Enter id    :  1
0. Enter Name  :  AK
0. Enter CGPA  :  82.01%

1. Enter id : 2
1. Enter CGPA : ~~AE~~ 82.5%
1. Enter Name : AL
```

Code

```
#include <stdio.h>
# include <conio.h>
void main()
{
        struct student
        {      int id;
               float CGPA;
               char name[10];
        };
        struct student s[20];
        int size, i;
        clrscr();
        printf("\n How many records you want to insert :");
        scanf("%d", &size);
        for (i=~~0~~1; i<=size; i++)
        {   printf("\n %d Enter id :", i);
            scanf("%d", e[i].id);
            printf("\n %f Enter CGPA :", i);
            scanf("%f", e[i].CGPA);
            printf("\n %c Enter name :", i);
            scanf("%c", e[i].name);
        }
       printf("\n\n students Data \n\n");
       printf("\t id \t CGPA \t name");
for ~~printf~~(i =~~0~~1; i<= size; i++)
       { printf("\t %d \t %f \t %c")
       }
}
return 0;
```

B] NAP Which will demonstrate the use of structure & function

Step1 - Start turbo C + application

Step2 - Decide the structure as struct Student.

Step3 - Initialize the struct student with two more variable (int ~~[illegible]~~ roll and char name[10])

Step4 - Now inside void main display and void main declare int i, struct student s[10]

Step5 - Use the for loop for entering details of student upto 2 students and not more than that

Step6 - Print the details of students

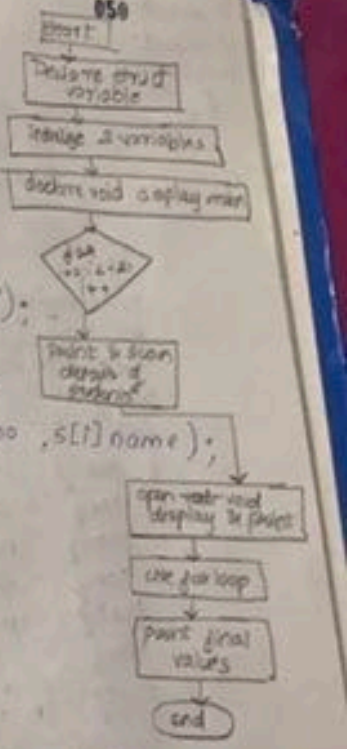
Step7 - Open void display again & put the values of the condition using printf.

Step8 - Terminate the program

Output -

```
Enter details of 2 student
Enter rollno & name - 22 Shreya
Enter rollno & name - 24 Alentina

RollNo - 22 Name - Shreya
RollNo - 24 Name - Shreya
```

```
~~#inde~~
#include <stdio.h>
struct student
{ int roll;
  char name[10];
};
void display(struct student s1[10]);
void main()
{ int i;
  struct student s[10];
  clrscr();
  printf("\n Enter details of 2 students");
  for(i=0; i<2; i++)
  { 
     printf("\n Enter roll & name ");
     scanf("%d %s", &s[i]; rollno ,s[i] name);
  }
  display(s);
  getch();
}
void display (struct student s1[10])
{ int i;
  printf("\n \n");
  for(i=0 ; i<2; i++)
       printf("\n Roll = %d \t Name %s", s1[i]roll, s1[i]name)
}
getch();
}
```



Practical - 9

A] Qn - WAP to copy one string into anothere string

Algo

Step1 - Input string from user and store it to some variable say text1

Step2 - Declare anothere variable to store copy of first string in text2

Step3 - Run a loop from 0 to end of string. The loop structure should be like
for(I= 0 ; text1[i] ; != '\0', i++)

Step4 - Inside the loop for each character in text1 copy to text2 Say text2[i] = text1[i]

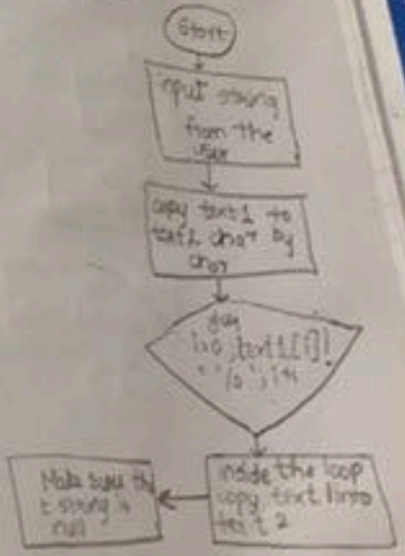
Step5 - Finally after loop make sure the copied string ends with NULL charactor i.e text 2[i] = '\0';

Output

```
Enter any string - There are 7 days in a week
1st string — There are 7 days in a week
Total string copied - 26
```

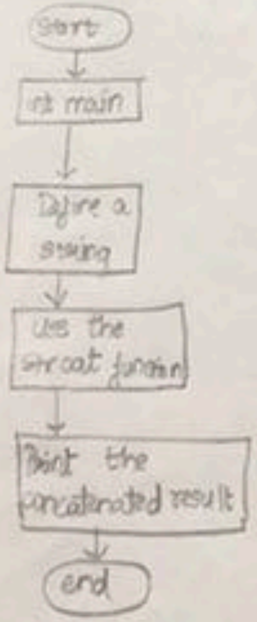
#Code

```
#include <stdio.h>
# define MAX_SIZE 100
int main()
{char text1 [MAX_SIZE];
 char text2 [MAX_SIZE];
  int i ;
 printf("Enter any string");
 gets(text1);
 for (I= 0 ; text1[i]!= '\0' ;i++)
  { text2[i] = text1[i];
  }
 text 2[i] = '\0';
 printf ("first string = %s\n", text1);
 printf ("first string copy = %s\n", text2).
 printf (" Total character copied = %d\n");
 return 0;

}
```


Start
input string from the user
copy text1 to text2 char by char
for I=0; text1[i]!='\0'; i++
inside the loop copy text1 into text 2
Make sure the string is null

Code

```
# include <stdio.h>
# include <string.h>
int main (int argc, const char* argv[ ])
 {
    char example [100];
    strcpy (example, "Rahul");
 };
Strcat (example, " is over 18");
Strcat (example, "years old");
printf ("%s \n", example);
return 0;
}
```



8 Write a program which will demonstrate the use of String library function

Strcat : The strcat() function will append a copy of the source string to the end of destination String. The Strcat function takes 2 arguments 1) dest 2) src

};

The strcat function returns a pointer (where the resulting concatenated string resides)

Output Rahul is over 18years old

C WAP which displays the length of a string without using string function

Alg-

Step1 - Take a string as input and store it in array

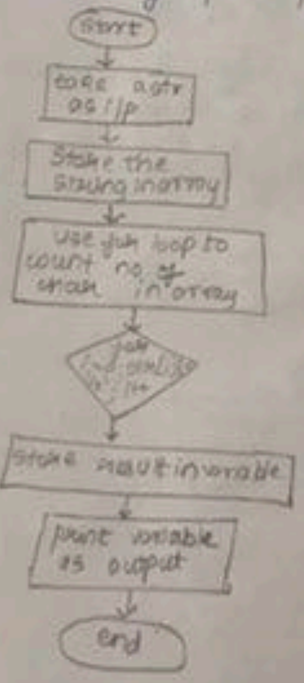
Step2 - Using for loop count the number of characters in the array and store the results in the variable.

Step3 - Print variable as output

Output - Enter a string
It is a cold night
The length of str is the no of characters in it so the length is 17.

Code

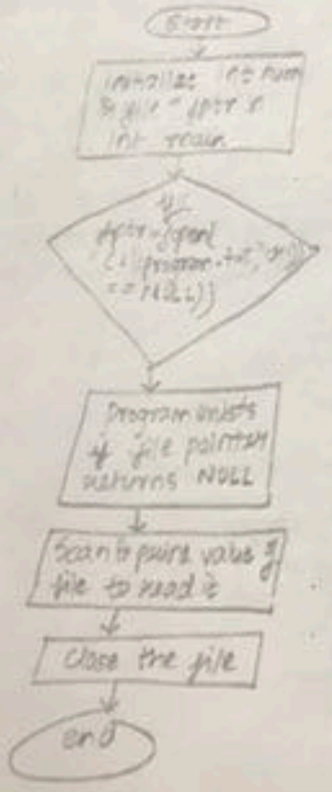
```
#include <stdio.h>
void main()
{ char string [50];
  int i , length = 0;
  printf ("Enter a string\n");
  gets (string);
  for (i=0 ; string[i] != '\0' ; i++)
  {   length ++;
  }
  printf ("The len of str is the no of characters in it\n");
  printf ("So the len of : %s = %d\n", string, length);
}
```



code for reading from a text file (opening/closing. Text file (
and its contents are 87, 88, 89, 90.

```
#include <stdio.h>
#include <conio.h>
int main ()
{  int num;
   FILE *fptr;
   if ((fptr = fopen("C:\\program.txt","r")) == NULL))
   {  printf("Error : opening file");
       exit(1);
   }
fscanf(fptr, "%d", &num);
printf("Values are = %d", num);
fclose(fptr);
return 0;
}
```

Flowchart →


Start
Initialize int num & file *fptr & int main
if (fptr = fopen(C:\\program.txt","r") == NULL))
Program exists if file pointer returns NULL
Scan & print value of file to read it
Close the file
end

Practical 10

Aim - Program for file open, file read and file close.

- f.open() → Opens a existing file / create a new file
- f.read() ⟶ Reads records from file
- f.close() ⟶ closes a file

output - values are - 87 88 89 90

B Aim - WAP for fgetc ();fgetchar (). & ptrchar () function

Algo/Description - fgetchar is a file handling function

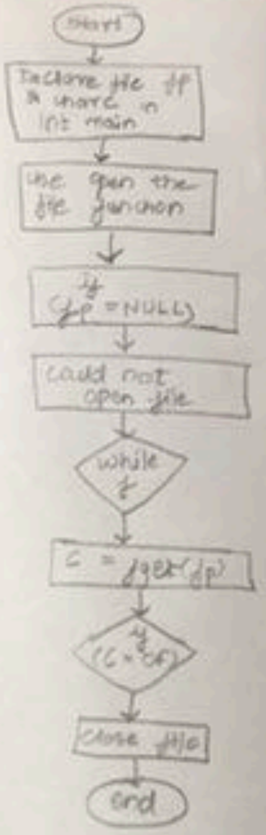
- It is used to read a single character from keyboard i/p

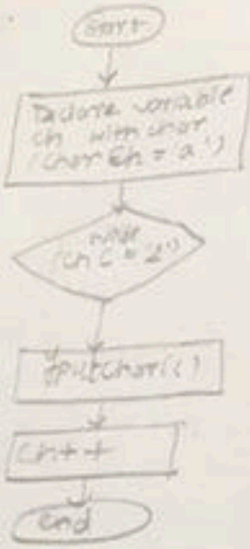
Code

```
#include <stdio.h>
# include < ctype.h>
int main ()
{
  char c ;
  printf ("Enter some character. Enter to exit \n
  While (c! = '$');
  {
   c = fgetchar ();
   printf ("\n Enter character is : ");
   ptrchar (c);
   printf ("\n ");
 }
  return 0;
}
```

Output -

```
Enter some character Enter $ to exit
A
Entered character is A
B
Entered character is B
$
```



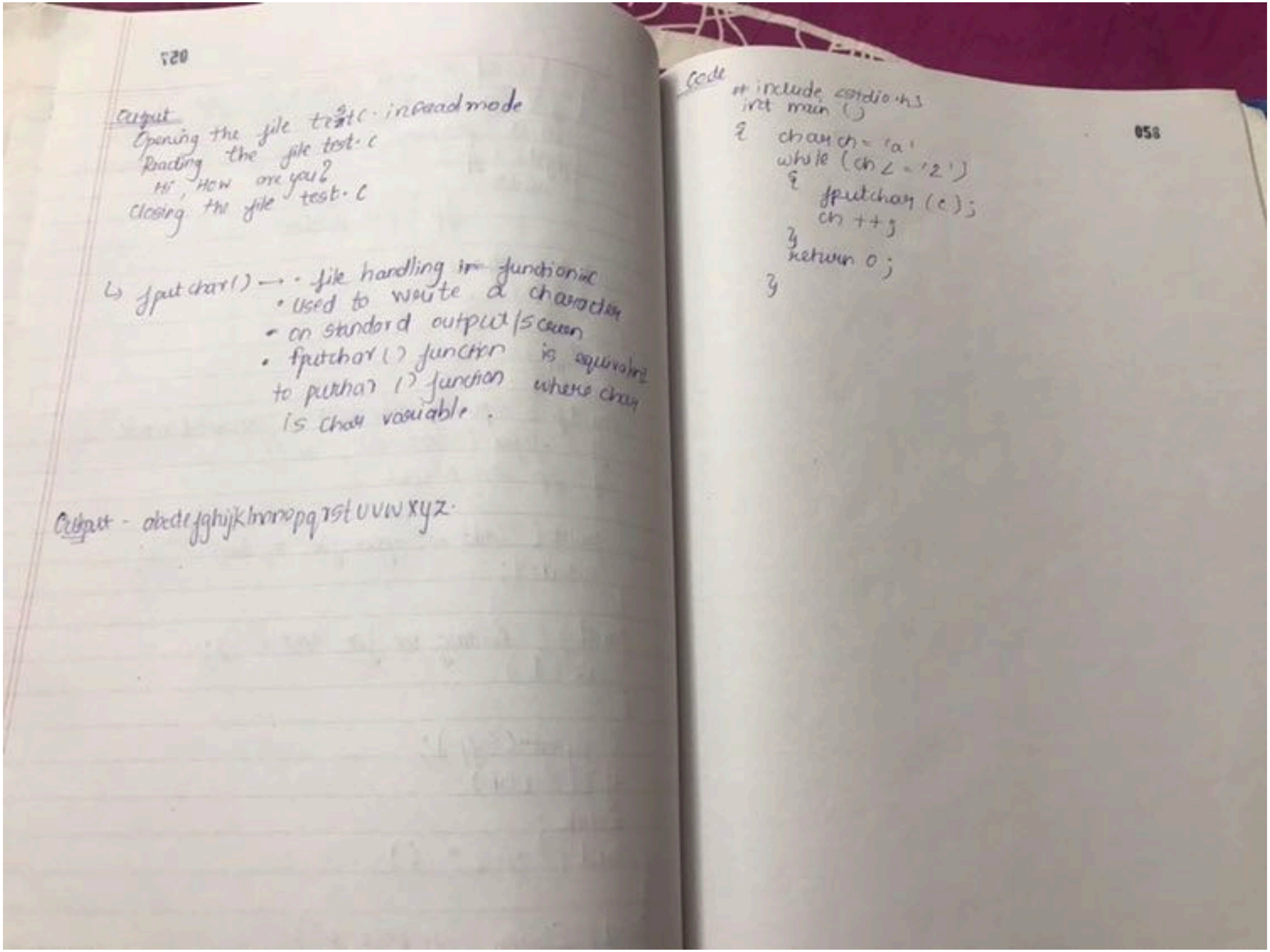


↳ fgetc () → Used to read a character from a file. Read single character at a time in a program we use fgetc ( ) function fgetc (fp)
where

fp = file pointer

<u>Code</u> :

```
#include <stdio.h>
int main ()
{
 file *fp;
 char c;
 printf ("Opening file test.c in read mode");
  fp = fopen ("test.c", "r")
  if ( fp == NULL )
 {
   print ("Could not open file of test.c");
   return 1;
 }
 print f ("Reading the file test.c");
 while (1)
 {
  c = fgegt c (fp);
  if (c = EOF)
  break;
 print ("%c", c)
}
print ("Closing file test c");
fclose (fp);
return 0
}
```



Output

```
Opening the file test.c in read mode
Reading the file test.c
Hi, How are you?
closing the file test.c
```

↳ fputchar() → · file handling in functionic
· used to write a character
· on standard output/screen
· fputchar() function is equivalent to putchar() function where char is char variable.

Output - abcdefghijklmnopqrstuvwxyz.

Code

```
#include <stdio.h>
int main ()
{   char ch = 'a'
    while (ch <= 'z')
    {
        fputchar (c);
        ch ++;
    }
    return 0;
}
```