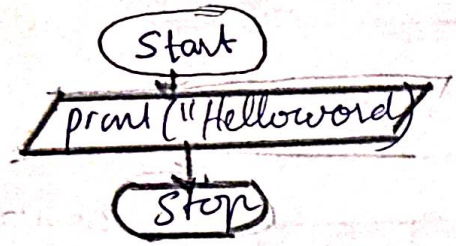
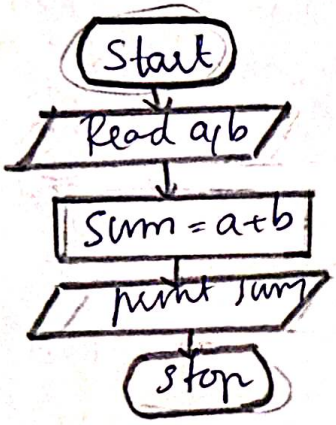
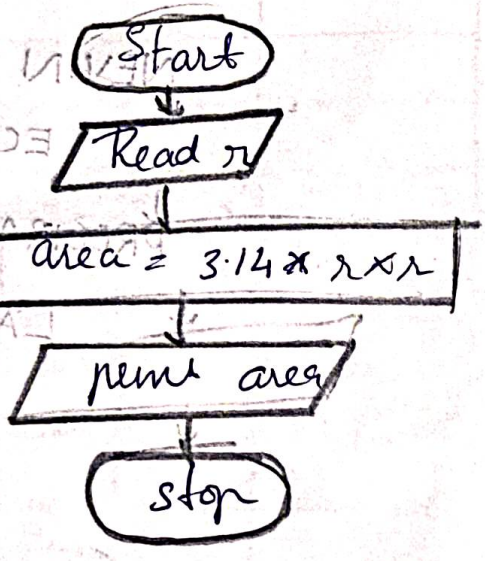
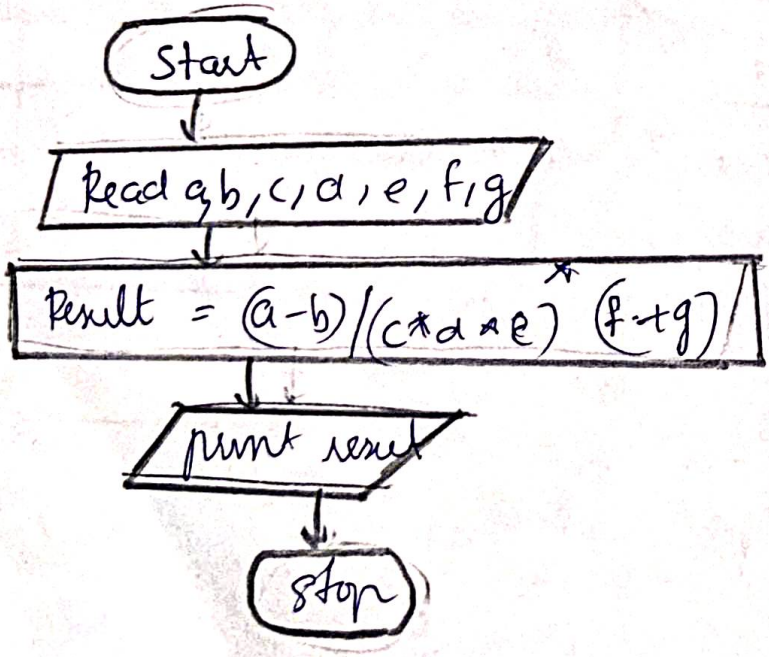
FLOWCHART


Start
print("Helloworld")
Stop


Start
Read a,b
Sum = a+b
print sum
stop


Start
Read r
area = 3.14 * r * r
print area
stop


Start
Read a,b,c,d,e,f,g
Result = (a-b)/(c*d*e)^(f+g)
print result
stop

EXPERIMENT-1

FAMILIARIZATION OF CONSOLE I/O IN C

Aim:
Familiarization of console I/o and operation in C.

(i) Display "Hello world".
(ii) Read two number, add them and display their sum.
(iii) Read the radius of a circle, calculate its area and display it.
(iv) Evaluate the arithmetic expression and display its solution (a-b/c*d+e)*(f+g)

PROGRAM

(i)
```
#include <stdio.h>
int main()
{
printf("Hello world");
return 0;
}
```

ii)
```
#include <stdio.h>
int main()
{
int a,b, sum;
a=2;
b=7;
sum=a+b;
printf("The sum of %od and %od ":"%od", a,b,sum)
```

OUTPUT

(i) Hello world

ii) The sum of 2 and 7 is :9

iii) Enter the radius : 10
area of the circle is 314

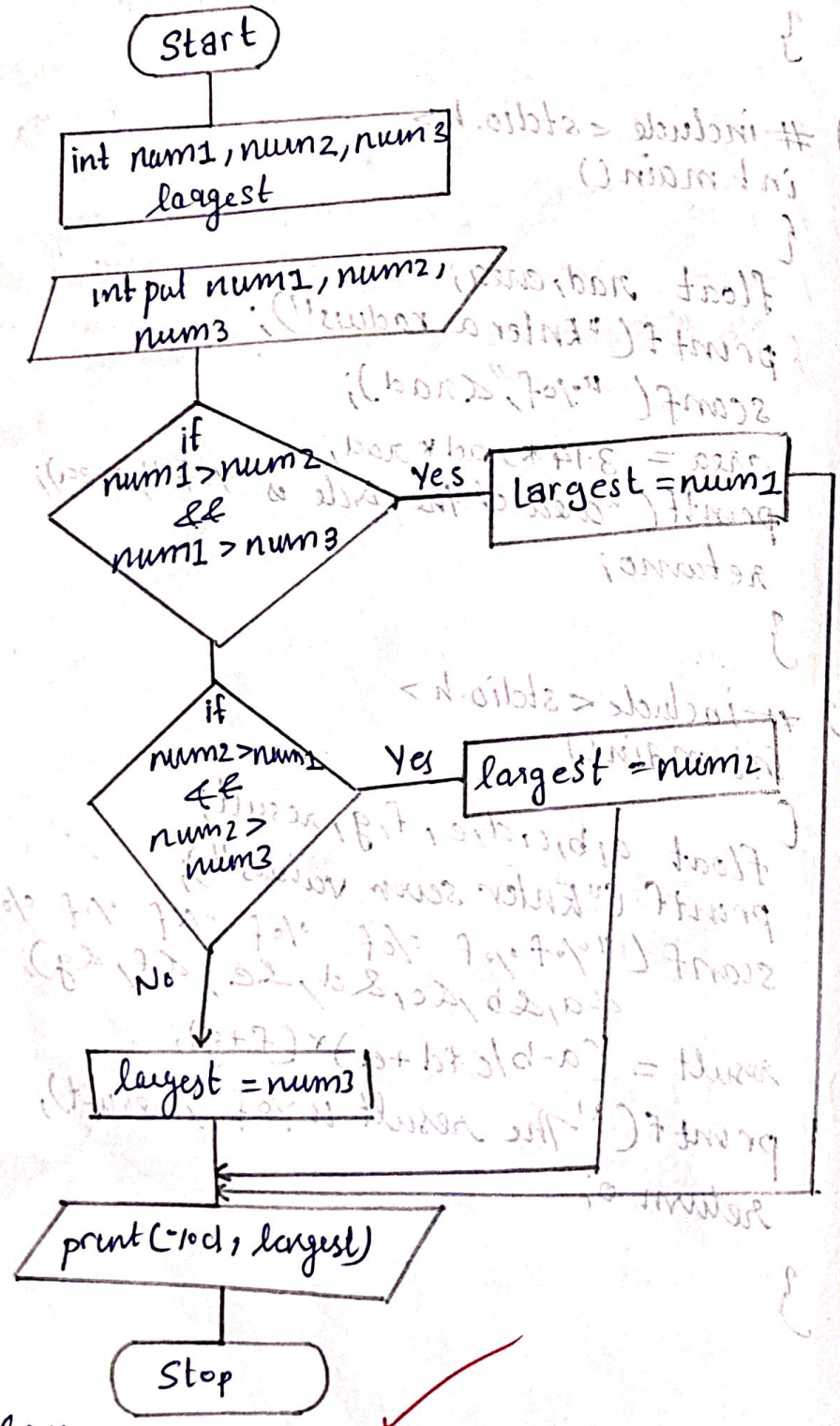
iv) Enter seven numbers 10 20 20 20 20 20 20
The result is 400.00

```
    return 0;
}
```

iii)
```
#include <stdio.h>
int main()
{
    float rad, area;
    printf("Enter a radius");
    scanf("%of", &rad);
    area = 3.14 * rad * rad;
    printf("area of the circle is %f", area);
    return 0;
}
```

iv)
```
#include <stdio.h>
int main()
{
    float a, b, c, d, e, f, g, result;
    printf("Enter sever values");
    scanf("%f %f %f %f %f %f %f",
          &a, &b, &c, &d, &e, &f, &g);
    result = (a-b/c*d+e)*(f+g);
    printf("The result is %f", result);
    return 0;
}
```

FLOWCHART



OUTPUT

Enter three numbers 10 20 30

The largest number is : 30

EXPERIMENT-2

LARGEST AMONG THREE NUMBERS

Aim:
Read 3 integer values and find the largest among them.

Program:

```
#include <stdio.h>
int main()
{
int num1, num2, num3, largest;
printf("Enter three numbers:");
scanf("%d %d %d", &num1, &num2, &num3);
if(num1>=num2 && num1>=num3)
{
largest = num1;
}
else if(num2>=num1 && num2>=num3)
{
largest = num2;
}
else
{
largest = num3;
}
printf("The largest number is : %d \n", largest)
return 0;
}
```

Result
program to find largest among three number is executed and famd sucessful

OUTPUT

Enter a number : 5
5 is a prime number.

Enter a number : 12
12 is not a prime number

EXPERIMENT-3

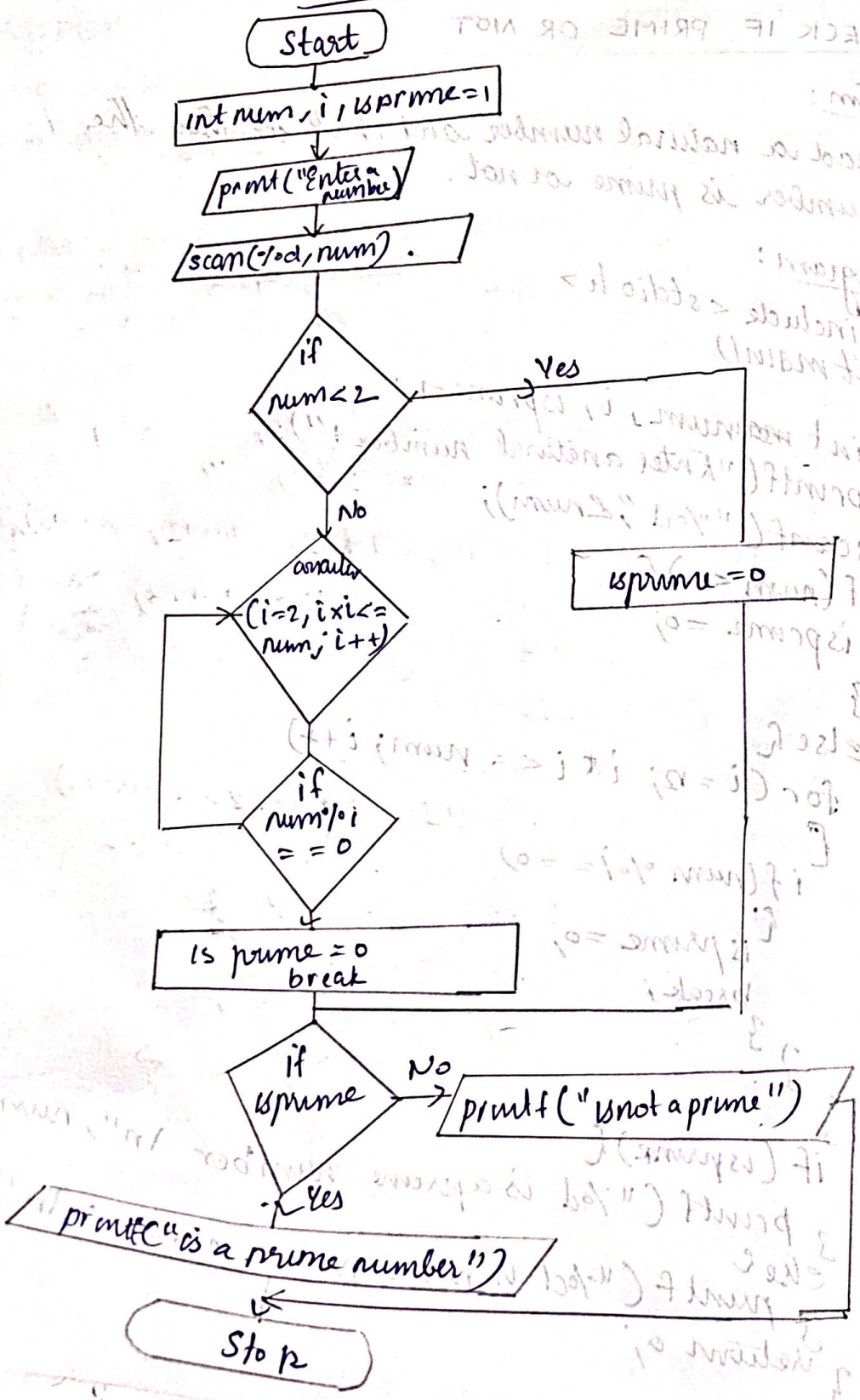
CHECK IF PRIME OR NOT

Aim:
Read a natural number and check whether the number is prime or not.

Program:

```
#include <stdio.h>
int main()
{ int num, i, isprime=1;
printf("Enter a natural number:");
scanf("%d",&num);
if(num<2){
 isprime=0;
}
else{
 for(i=2; i*i<=num; i++)
 {
  if(num%i==0)
   {
    isprime=0;
    break;
   }
 }
}
if(isprime){
 printf("%d is a prime number\n", num);
}
else{
 printf("%d is not a prime number\n", num);
}
return 0;
}
```

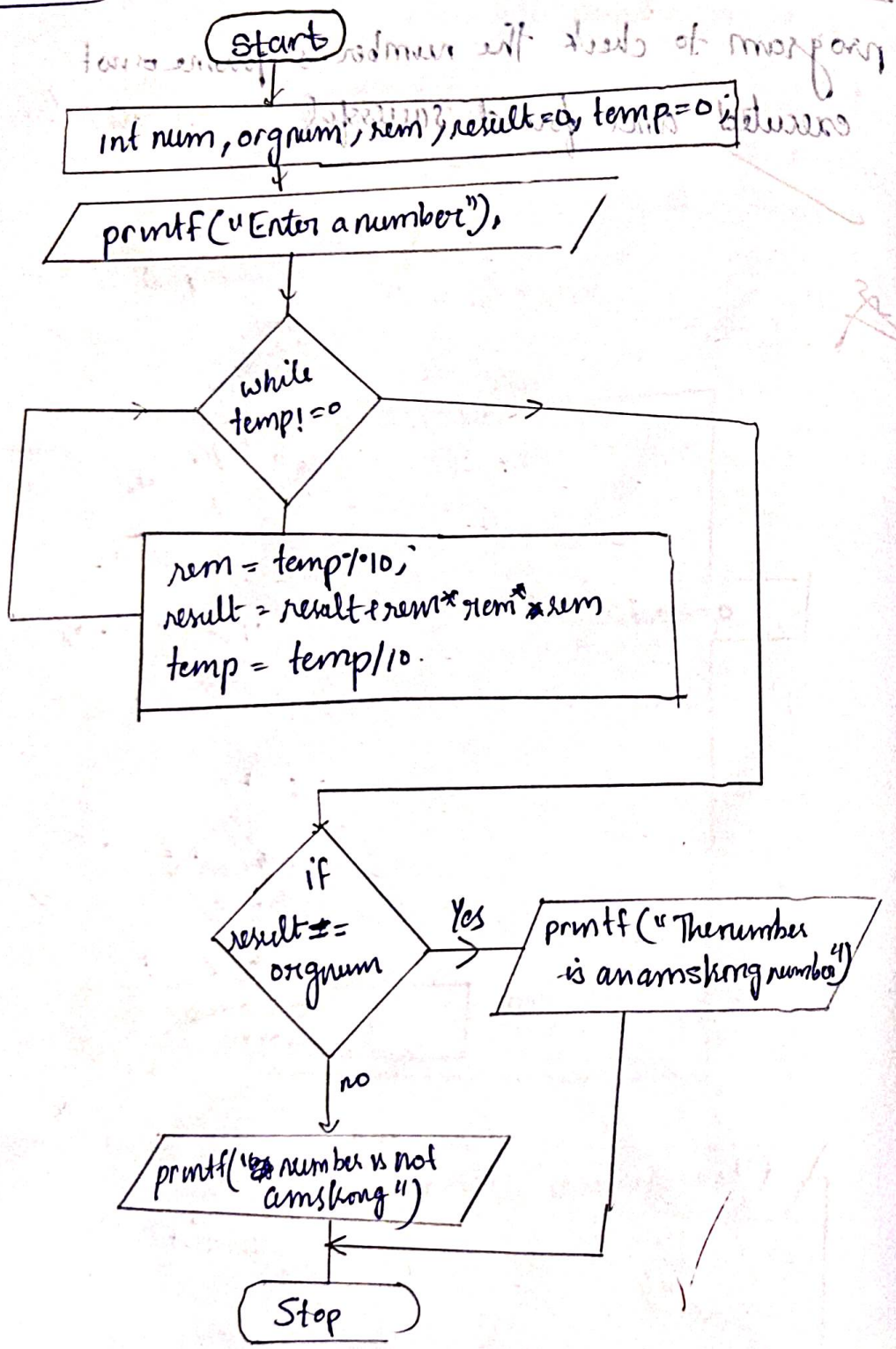
FLOWCHART


Start
int num, i, isprime=1
print("Enter a number")
scan("%d", num)
if num<2
Yes
No
isprime=0
for (i=2, i*i<=num; i++)
if num%i==0
is prime=0 break
if isprime
No
printf("is not a prime")
Yes
printf("is a prime number")
Stop

Result

program to check the number is prime or not executed and found successful

FLOWCHART


Start
int num, orgnum, rem, result=0, temp=0;
printf("Enter a number");
while temp!=0
rem = temp%10;
result = result+rem*rem*rem
temp = temp/10.
if result== orgnum
Yes
printf("The number is an armstrong number")
no
printf("number is not armstrong")
Stop

EXPERIMENT NO:4

CHECK IF AMSTRONG NUMBER OR NOT

Aim: Read a number check if the number is amstrong or not

Program:

```
#include<stdio.h>
int main()
{
 int num, orgnum, rem, result=0, temp=0;
 printf("Enter a number");
 scanf("%d", &num);
 orgnum = num;
 temp = num;
 while(temp!=0)
 {
  rem = temp%10;
  result = result + rem*rem*rem;
  temp = temp/10;
 }
 if(result==orgnum)
 {
  printf("%d is an amstrong number \n", num);
 }
 else
 {
  printf("%d is not an amstrong number \n", num);
 }
```

output

enter a number 153

153 is an armstrong number

enter a number 22

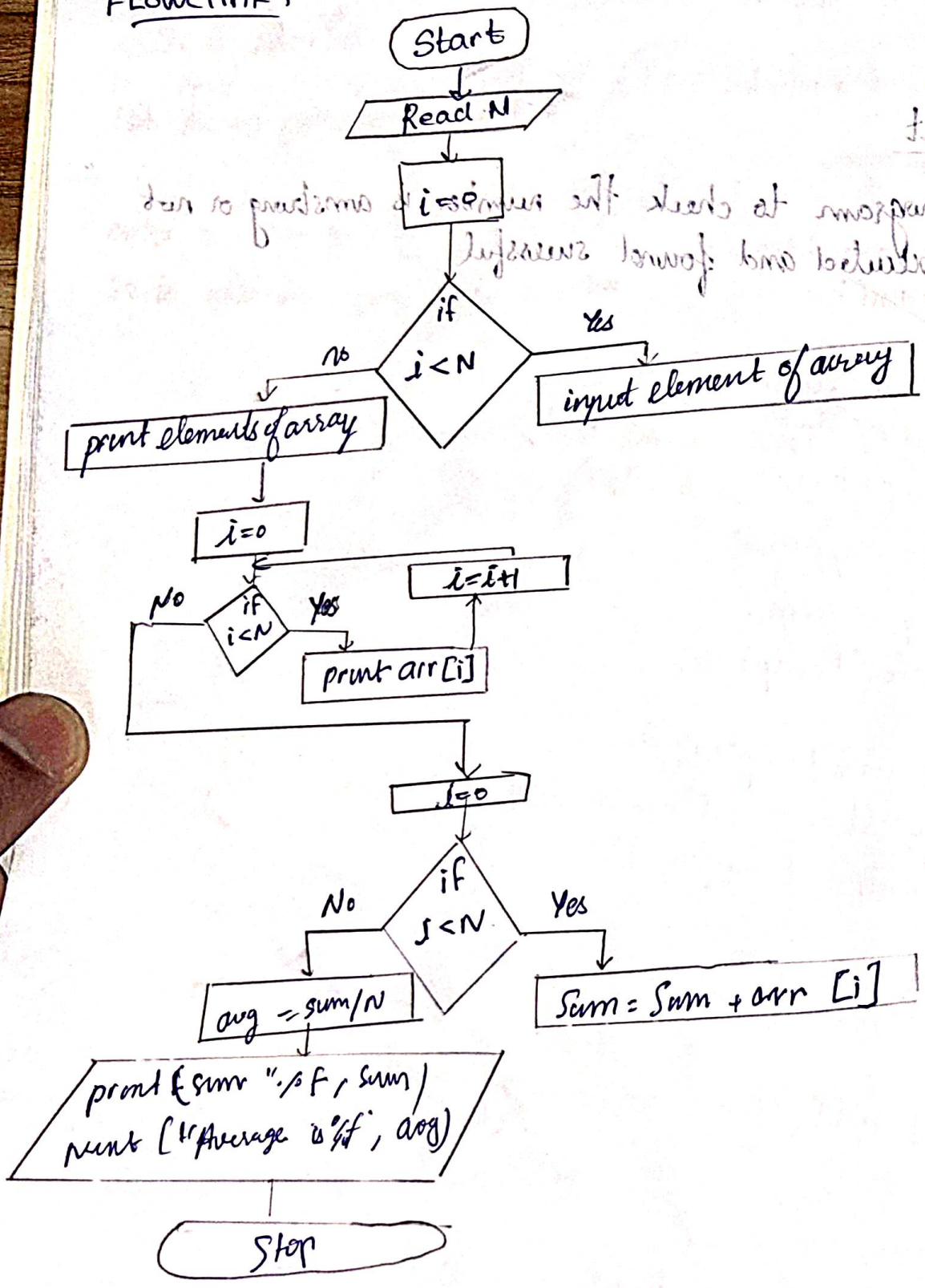
22 is not an armstrong number

return 0;
}

Result

The program to check the number is amstrong or not is executed and found sucessful

FLOWCHART


Start
Read N
i=0
if
i<N
Yes
input element of array
No
print elements of array
i=0
i=i+1
No
if
i<N
Yes
print arr[i]
j=0
if
j<N
No
Yes
avg = sum/N
Sum = Sum + arr[i]
printf("sum = %f", sum);
printf("\nAverage is %f", avg);
Stop

EXPERIMENT NO: 5

FIND THE SUM AND AVERAGE OF ELEMENTS IN AN ARRAY

Aim : Read n integers store them in an array find their sum and average.

Program

```
#include<stdio.h>
int main()
{
float a[100], sum=0, avg;
int i,n;
printf("Enter the syz of array: ");
scanf("%d",&n);
printf("Enter the elements of the array: \n");
for(i=0;i<n;i++)
{printf("a[%d] - ",i);
 scanf("%f",&a[i]);
}
for(i=0,i<n;i++)
{
 sum=sum+a[i];
}
avg=sum/n;
printf("Sum is %f \n",sum);
printf("Average is %f",avg);
return 0;
}
```

OUTPUT

Enter the size of the array : 5
enter the elements of the array
a[0] : 1
a[1] = 2
a[2] = 3
a[3] = 4
a[5] = 5

the sum of the elements in the array is 15.00

the average of the elements in the array is

Result

The program to read n elements in array and to find sum and average is executed and found successful.

ALGORITHM

1. Start
2. Read num
3. Using for loop, iteratively read elements of array num times.
4. initialize index = -1
5. Read search.
6. Using for loop, check if element of array is equal to search starting from first element num times
7. If an element becomes equal to search index = index of the element and break out of the loop
8. If index is equal to -1, search does not exist in array
9. else print, "search" is at index : index + 1
10. Stop

EXPERIMENT NO: 6

FIND AN ELEMENT OF ARRAY USING LINEAR SEARCH

Aim: Read n integers, store them in an array and search for an element in the array using an algorithm for linear search.

Program:

```
#include<stdio.h>
int main()
{
  int array[100], search, c, num;
  printf("Enter the number of elements in array\n");
  scanf("%d",&num);
  printf("Enter %d numbers\n", num);
  for (i=0; i<num; i++)
     scanf ("%d",&array[i]);
  printf("Enter the number to search\n");
  scanf("%d",&search);
  for (i=0; i< num; i++)
  {
     if(array[i]==search)
     {
        printf("%d is present at location %d\n",
               search, i+1);
        break;
     }
  }
  if (i==number)
  printf ("%d is not present in array\n",search);
} return 0;
```

OUTPUT

Enter the no of elements : 5

enter 5 elements:

1

2

3

4

5

enter the element to search : 4

element 4 found at position 4

Result

The program to read a integers on array and to search element is executed and found successful

ALGORITHM

1. Start
2. Declare integer n, j, i, swap and array[100]
3. input value of n
4. for i<n; input value for array[i](i=0)
5. Repeat step4 until i=n
6. Initialise i and j equal to zero
7. for i<n-1 and j<n-j-1; if array[j]>array[i+1]
8. print the sorted array
9. Stop.

EXPERIMENT – 7

BUBBLE SORT

AIM:
To read n integers, store them in an array and elements using bubble sort algorithm

PROGRAM

```
#include <stdio.h>
int main()
{
int n,j,i,array[100],swap;
printf("Enter no of elements :\n");
scanf ("%d",&n)
printf ("Enter %d integers :\n",n);
for (i=0;i<n;i++)
{scanf ("%d",&array[i]);
}
for(i=0;i<n-1;i++)
{
  for(j=0;j<n-i-1;j++)
   {
    if(array[j]>array[j+1])
      [
      swap = array[j];
      array[j] = array[j+1];
      array[j+1] = swap;
      }
}
}
```

OUTPUT

```
enter the no of elements
5
enter the integers:
1
45
23
5
7
sorted list in ascending order:
1
5
7
23
45
```

```c
printf("Enter the sorted list in ascending order:")
 for(i=0; i<n ;i++)
   printf("%d \n",array[i]);
 relurn 0;
}
```

Result

The program to read n Integers and store them using bubble sort algorithm is executed and found succesfull

17/12/2025

ALGORITHM

1. Start
2. Include <stdio.h> and <string.h>
3. Declare character, char string[20], arr[100];
4. Declare integer, int i, length, flag=0
5. Input the string
6. Find the length of string using strlen.
7. Copy string from string to arr using strcpy
8. Initialise j=0
9. For i<length/2; if arr[i] not equal to arr[length-j]
   for all i = length/2
10. Print "string is not palindrome"
11. Else, print "string is palindrome"
12. Stop.

EXPERIMENT - 8

PALINDROME

AIM

To read a string, store it in an array and check if palindrome or not

PROGRAM

```
#include <stdio.h>
#include <string.h>
int main()
{ char string[20], arr[100];
  int i, length, flag = 0;
  printf("Enter a string : ");
  scanf ("%d", string);
  length = strlen (string);
  strcpy (arr, string)
  for (i=0; i < length/2; i++)
  {
    if(arr[i] != arr [length-i-1])
    flag = 1;
    break;
  }
  if(flag)
  {printf("%0s is not a ~~plin~~ palendrome \n", arr);
  }else
```

OUTPUT

Enter the string:
MALAYALAM

MALAYALAM is a palindrome

Enter the string
Red
Red is not a palindrome

```
{printf("%s is a palindrome \n",arr);
}
return 0;
}
```

RESULT

The program to check palindrome is executed and found succesful.

ABhishek A.
12/12/2025

ALGORITHM

1. Start
2. include <stdio.h>
3. Declare characters str1[100], str2[100], final[200]
4. Declare integers i=0, j=0, k=0, n=0
5. Input str1 and str2
6. While str[i] and str2[j] != $
   increment them by i++ and j++ respectively
7. for both n<i and n<j, final[k++] = str1[n] and final[k++] = str2[n] respectively
8. Repeat until i=n and j=n
9. Print concatenated string
10. Stop.

EXPERIMENT-9

STRING CONCATENATION

AIM

To read two strings (each ending with $ symbol) store them in arrays and concatenate them with library function

PROGRAM

```
#include <stdio.h>
int main()
{
  char str1[100], str2[100], final[200];
   int i=0, j=0, k=0;
printf("Enter the first string (ending with $):");
fgets(str1, sizeof(str1), stdin);
printf("Enter the second string (ending with $):");
fgets(str2, sizeof(str2), stdin);
while(str1[i] != '$')
{
   i++;
}
str1[i] = '\0';
while(str2[j] != '$')
   {
      j++
   }
```

EXPERIMENT - 9

AIM

OUTPUT

enter the first string (ending with $) : reverse$
enter the second string (ending with $) : sunday$
concatenated string : reversesunday

```
str2[j] = '\0';
for (int n=0 ; n<i ; n++)
{
final[k++] = str1[n];
};
for (int n=0; n<j; n++)
{final[k++] = str2[n];
}
final[k] = '\0';
printf("concatenated string : %s \n", final);
return 0;
}
```

RESULT

The program for string concatenation is executed and found successful

17/12/2024

ALGORITHM

1. Start
2. Include < stdio.h > and < clyne.h >
3. declare char = line [150].
4. Declare int vowels, constanants, digits and spaces
5. Initialise vowels = consonants = digits = space = i = 0
6. Input string
7. Transform uppercase to lowercase with to lower.
8. For line [i] check vowels a, e, i, o, u. if found ++vowels
9. for line[i] = 'a' and line [i] <= z found ++digit
10. for line [i] >= '0' and line [i] <= '9' if found ++digit
11. for line [i] = ' ' if found ++space
12. print vowels, consonants, digits, spaces.
13. Stop

EXPERIMENT - 10

Vowels consonents and digits

AIM

Read a string (ending with $ symbol) store it in an array and count the number of vowels, consonents digits and spaces in them.

PROGRAM

```
#include <stdio.h>
~~#include <ctype.h>~~
int main()
{
char line[150];
int vowels, consonants, digits, spaces, i=0;
vowels = consonants = digits = spaces = 0;
printf("Enter string:");
fgets(line, sizeof(line), stdin);
while(line[i] != '$')
{
  i++;
}
for(int i=0; line[i] != '\0'; i++)
{
~~line[i] = tolower(line[i]);~~
if(line[i]=='a' || line[i]=='e' || line[i]=='i'
    || line[i]=='o' || line[i]=='u' || line[i]=='A'
    || line[i]=='E' || line[i]=='I' || line[i]=
    =='O' || line[i]=='U')
  {
  ++vowels;
}
```

EXPERIMENT - 10

OUTPUT

enter a string :
computer
vowels : 3
consonants : 5
digits : 2
whitespace : 0

```
else if [line[i]>='a' && line[i]<='z'||line[i
  {
   ++consonants;
  }
  else if (line[i]>='0' && line[i]<='9')
  {
   ++digit
  }
  else if (line[i]==' ']
   {
    ++space
   }
 }
printf("vowels:%d", vowels);
printf("\nconsonants:%d", consonants);
printf("\n digits :%d ", digits);
printf("\n white spaces:%d ", spaces);
 return 0;
}
```

RESULT

Program to count the numbers of vowels, consonants, digits and whitespaces is executed and found successful

Abhishek.A
17/2/2025