

Page # 1

Name : Janzaid Khalid

ID : 15386

Semester : 3rd

Program : BS (SE)

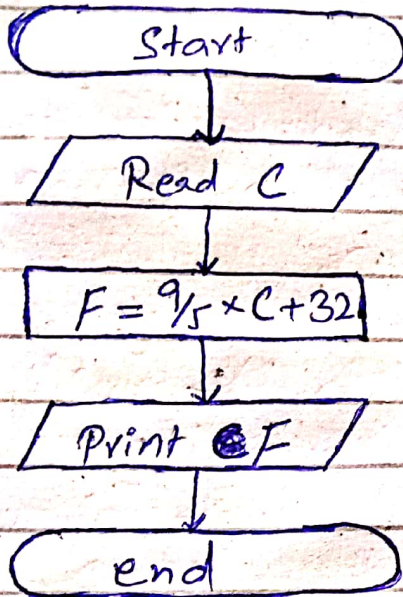
Paper : Programming fundamental

Teacher : Dr. Fazal-e-Malik.

Date : 25/08/2020

Q1 b) Solution :-

Flowchart :



Algorithm :-

Step 1 : Start

~~Step 2 : Read Celsius~~

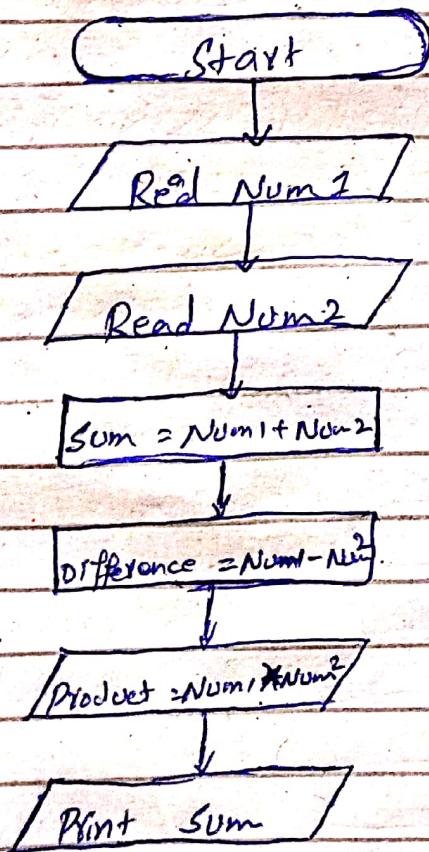
Step 2 : Read Temperature in Celsius C

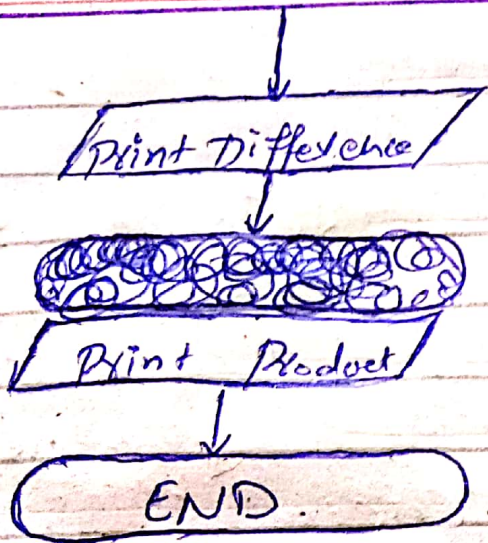
Step 3 :  $F \leftarrow \frac{9}{5} \times C + 32$

Step 4 : Print Temperature in Fahrenheit F.

Step 5 : End.

Q1(a) :- Solution :-





∴ Display on Screen :-

Please enter the first Number  
9

Please enter the second Number  
10

Sum = 19

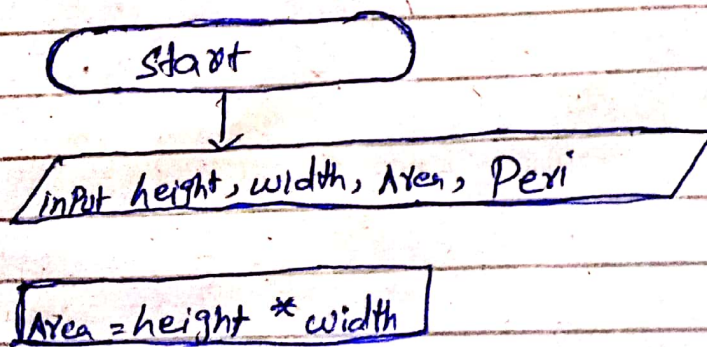
Diff = -1

Product = 90

Q29 :- Solution :-

```
#include <iostream>
using namespace std;
int main()
{
    int width, height, area, Peri;
    cout << "\n\n Find the area and Perimeter
    of a Rectangle : \n";
    cout << " _____ \n";
    cout << " Input the length of the rectangle : \n";
    cin >> height;
    cout << " input the width of the rectangle : \n";
    cin >> width;
    area = (height * width);
    Peri = 2 * (height * width);
    cout << " The area of the rectangle is : " <<
    area << endl;
    cout << " The Perimeter of the rectangle
    is : " << Peri << endl;
    cout << endl;
    return 0;
}
```

-: Flow chart :-



Page # 5



$$P_{vi} = 2 * (\text{height} * \text{width})$$

Print Area,  $P_{vi}$

END

Q3a) Discuss different types of Programming languages.

There are two major types of Programming languages.

- 1) Low Level Languages
- 2) High Level Languages

Low Level languages are further divided into Machine language and Assembly Language.

### Machine Language:-

Machine Language is the only language that is directly understood by the computer. It does not need any translator program.

- 1) The only advantage is that program of machine language run very fast.
- 2) There is nothing "below" machine language (only hardware).
- 3) Impossible for human to read. consist of only 0's and 1's.

→ 000100111110000

- 4) In the earliest days of computers, the only programming languages available were machine languages. Each computer had its own machine language, which was made of streams of 0s and 1s.

## Assembly languages :-

- 1) The next evolution in Programming came with the idea of replacing binary code for instruction and addresses with symbols. Because they used symbols, these languages were first known as symbolic languages. The set of these mnemonic languages were later referred to as assembly languages.
- 2) it is the first step to improve the programming structure, you should know that computer can handle numbers and letters.
- 3) The set of symbols and letters forms the Assembly language and a translator program is required to translate the Assembly language to machine language.
- 4) This translator program used for Assembly language is called Assembler.
- 5) To program in assembly you need to understand concept behind machine language and execution-fetch cycle of CPU.
- 6) Assembly is a machine-specific language.
- 7) Although Assembly and machine language might look similar, they are in fact two different types of languages.
  - > Assembly consists of both binary and simple words
  - > Machine code composed only of 0's and 1's.

## High Level Languages :-

- 1) Although assembly languages greatly improved programming efficiency, they still required programmers to concentrate on the hardware they were using. Working with symbolic language was also very tedious, because each machine instruction had to be individually coded. The desire to improve programmer efficiency and to change the focus from the computer to the problem being solved led to the development of high level languages.
- 2) Assembly language and machine level languages require deep knowledge of computer hardware while as in higher language you have to know the only the instructions in English words and logic of the problem.
- 3) Higher level languages are simple languages that use English and mathematical symbols like +, -, %, / etc. for its program construction.
- 4) For example COBOL (Common Business oriented language), FORTRAN (Formula Translation) and BASIC (Beginners All Purpose Symbolic Instruction Code) are high level languages.



## Page #9

Q3 b) How many translators are there to translate higher level language to machine language? Discuss.

Ans:-

There are two types of translators

- 1) Compiler
- 2) Interpreter

### Compiler :-

it is a program translator that translates the instruction of a higher level languages to machine language.

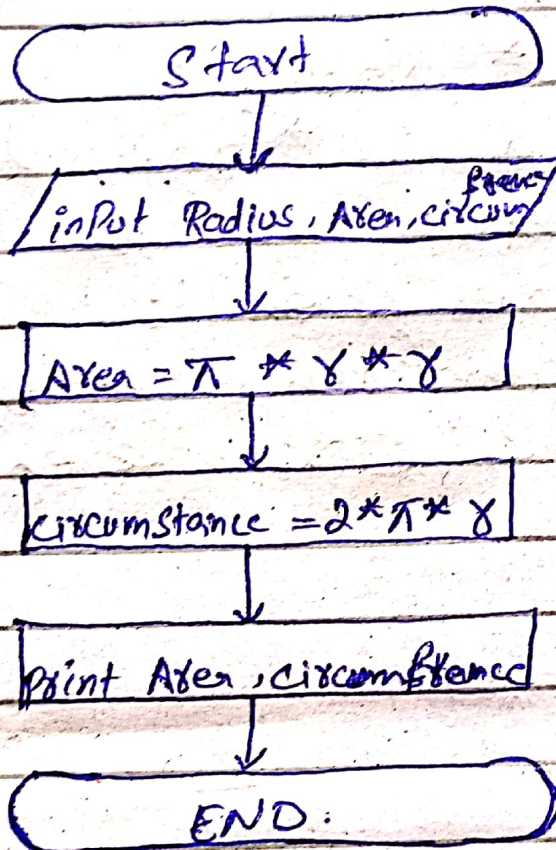
- 2) it is called compiler because it compiles machine language instructions for every program instructions of higher level language.
- 3) Thus compiler is a program translator like assembler but more sophisticated. it scans the entire program first and then translates it into machine code.
- 4) The program written by the programmer in higher level language is called source program. After this program is converted to machine languages by the compiler is called object program.
- 5) A compiler can translate only those source programs, which have been written, in that language.

## Interpreter :-

- 1) An interpreter is another type of program translator used for translating higher level language into machine language.
- 2) It takes one statement of higher level language, translate it into machine language and immediately execute it.
- 3) Translation and execution are carried out for each statement.
- 4) It differs from compiler, which translate the entire source program into machine code.
- 5) The advantage of interpreter compared to compiler is its fast response to ~~the~~ changes in source program do not require large memory in computer.
- 6) Thus compiled machine language program runs much faster than an interpreted program.

Q2b :- Solution :-

∴ Flow chart :-



-: Code :-

```
#include <iostream>
using namespace std;
int main()
```

```
{
```

```
int Radius, Area, circumference, Pi = 3.14;
cout << "Enter Radius" << cin >> Radius;
Area = Pi * Radius * Radius;
circumference = 2 * Pi * Radius;
cout << Area << circumference;
```

```
}
```