

Name Younas Khan
ID 12422
Subject Programming fundamentals

QNO1 a):-

```
#include <iostream>
#include <conio.h>
```

```
main()
```

```
{
```

```
int hours, payrate, grosspay;
```

```
cout << "How many hours did you work?"
```

```
cin >> hours;
```

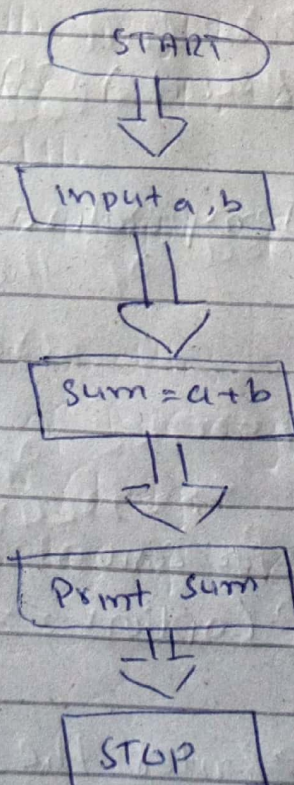
```
cout << "How much do you get pag
```

```
cin >> payrate;
```

```
cout << "your gross pay is\n", grosspay;
```

```
system("pause");
```

```
}
```



(2)

QNO 1 ⇒ (b)

```
#include <iostream>
using namespace std;
int main()
{
    float fahr, cel;
    char option;
    cout << "Choose from following option:" << endl;
    cout << "1. Celsius to Fahrenheit" << endl;
    cout << "2. Fahrenheit to Celsius" << endl;
    cin >> option;
    // option for converting Celsius into Fahrenheit
    if (option == '1')
    {
        cout << "Enter the temperature in Celsius:";
        cin >> cel;
        fahr = (1.8 * cel) + 32.0; // Temperature conversion formula
        cout << "\nTemperature in degree Fahrenheit:"
        << fahr << " F" << endl;
    }
    // option for converting Fahrenheit into Celsius
    else if (option == '2')
    {
        cout << "Enter the temperature in
        Fahrenheit:";
        cin >> fahr;
    }
}
```


(3)

```
cel = (fahr - 32) / 1.8; // Temperature  
conversion formula  
cout << "In Temperature in degree celsius:"  
<< "C" << endl;
```

}

else

```
cout << "Error wrong input." << endl;  
return 0;
```

}

Q No 2: (a) #include <iostream>

using namespace std;

int main()

{

int width, height, area, perimeter;

cout << "Enter width of Rectangle = ";

cin >> width;

cout << "Enter Height of Rectangle = ";

cin >> height;

area = height * width;

cout << "Area of Rectangle = " << area << endl;

perimeter = 2 * (height + width);

cout << "Perimeter of rectangle are = " <<

perimeter << endl;

return 0;

}

(4)

```
QNO 2: (b)
QNO 2: (b) #include <iostream>
# define PI 3.14159
using namespace std;
```

```
int main ( )
```

```
{
```

```
float radius, area, Circum;
```

```
cout << "n/n find the area and circum-  
ference of any circle: |n";
```

```
cout << "n/n find the area and circumference  
of any circle: |n";
```

```
cout << "input the radius (1/2 of diameter)  
of a circle: ";
```

```
cin >> radius;
```

```
Circum = 2 * pi * radius;
```

```
area = pi * (radius * radius);
```

```
cout << "The area of the circle is: " <<  
<< area << endl;
```

```
cout << "The Circumference of the  
circle is: " << Circum << endl;
```

```
cout << endl;
```

```
return 0;
```

```
}
```


5

Output

Input the radius (r) or diameter of a circle is: 5

The area of the circle is: 78.

The circumference of the circle is: 31.4159.

① No 3 (a) Programming languages, programming languages specially developed so that you could pass your data and instructions to the computer to do specific job.

These are two major types of programming languages.

① Low level language

② High level language

Low level language are further divide into machine languages and Assembly language.

High level language are, for scientific application Fortran and C languages are used on the other hand COBOL is used for business applications.

(6)

Machine language:

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

The only advantage is that program of machine language run very fast.

There is not "below" machine language only hardware.

Impossible for humans to read
Consists of only 0's and 1's.

000100111110000

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 Language

The next evolution in programming came with the idea of replacing binary code for instruction and address with symbols. Because they used symbols, these language were first known as symbolic language. The set of these mnemonic

(7)

language were later referred to as assembly languages.

It is the first step to improve the programming structure, you should know that computers can handle numbers and letters.

The set of symbols and letters from the assembly language and a translator program is required to translate the assembly language to machine language.

This translator program is used for assembly language is called Assembler.

To program in assembly you need to understand concepts behind machine language and execution fetch cycle of CPU.

Assembly is a machine specific language.

Although Assembly and Machine language might look similar, they are in fact two different languages.

> Assembly consists of both binary and simple words.

> Machine code composed only of 0s and 1s.

(2)

High Level Language :-

Although assembly language 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.

Assembly and machine level language required deep knowledge of computer hardware where as in higher language you have to know only the instruction in english words and logic of the problem.

Higher level languages are simple language that use english and mathematical symbols like +, -, %, for its program construction.

Any higher level language has to be converted to machine language for the computer to understand.

(9)

For example COBOL (Common Business language), FORTRAN (Formula Translation) and BASIC (Beginner All-purpose instruction code) are high level languages.

Advantages of High Level Languages.

Higher level language have a major advantage over machine and assembly language that higher level language are easy to learn and used (similar to the language used by us in our day to day life).

~~Compiler~~

Q NO (3) (b) = Compiler:-

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

It is called compiler because it compiles machine language instruction for every programs instruction of higher level language.

Thus compiler is a program translator

(10)

Like assembles but more sophisticated it scans the entire program first and then translate it into machine code.

The program written by the programmer in higher level language is called source program. After this program is converted to machine language by the compiler it is called object program.

A compiler can translate only those source programs, which have been written, in that language.

X Interpreter :- An interpreter is another type of program translator used for translating higher level language into machine language. It take one statement of higher level language, translate into machine language and immediately execute it.

Translation and execution are carried out for each statement. It differs from compiler, which translate the entire source program into machine code.

(11)

The advantage of interpreter compared to compiler is its fast response to changes in source program do not require large memory in computer.

The disadvantage of interpreter is that it is time consuming method because each time a statement in a program is executed then it is first translated.

Thus compiled machine language program runs much faster than an interpreted program.