**Session 6 (unit 4): Friend functions and this keyword and Access Specifiers**

|  |  |
| --- | --- |
| Roll No. A016 | Name: Varun Khadayate |
| Program: B-Tech (CSBS) | Division: SY |
| Batch: 1 | Date of Submission: 18-09-2020 |

1. **How a member function of a class can acts as a friend of another class.**

**Ans:** We can access function of one class as a friend of another class by same way as we make a function as a friend of a class. But, here is the catch, we have to add **class\_name::** instead of **class\_name:** declaration before the name of that function in the class whose friend it is being declared.

**CODE:**

#include<iostream>

#include<string>

using namespace std;

class roll\_no;

class final\_data

{

public:

void display(roll\_no obj);

};

class roll\_no

{

int x;

public:

roll\_no()

{

x = 16;

}

friend void final\_data::display(roll\_no);

};

void final\_data::display(roll\_no obj)

{

cout << obj.x << endl;

}

int main()

{

roll\_no rn;

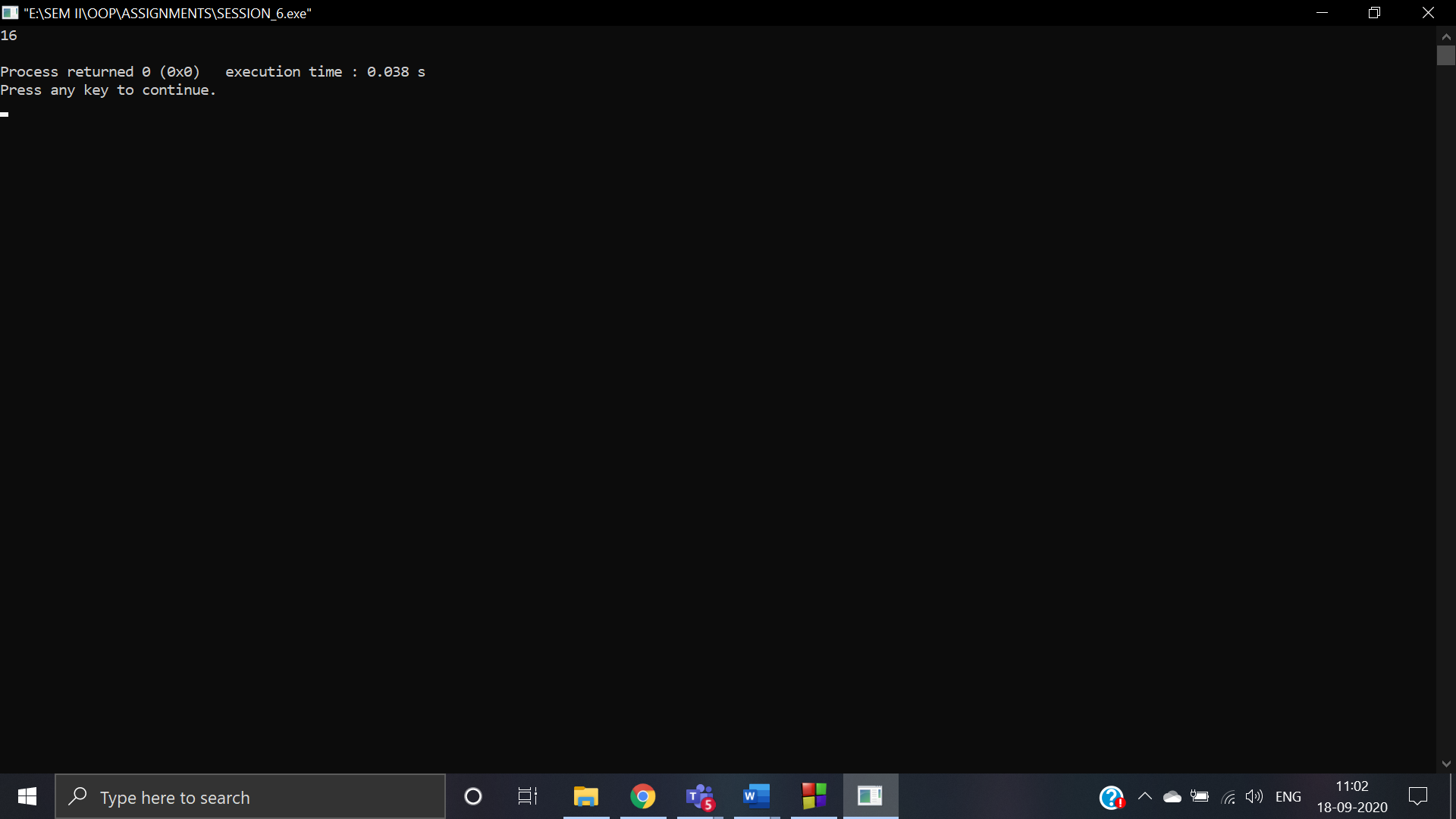
final\_data fd;

fd.display(rn);

return 0;

}

**OUTPUT:**



1. **State with the help of an example how ‘this’ keyword can be used in place of constructors in a class. Take an example of entering the student’s details and his marks in three subjects and displaying them using this keyword.**

**ANS:** “this” keyword is basically a pointer that points to the current instance (object) of the class. Hence, this is known as this Pointer in C++. There are following uses of this keyword in the C++:

* Access the currently executing object of a class.
* Access the data members of the currently executing object.
* Calling the member functions associated with the currently executing object.
* To resolve the shadowing issue, when a local variable has a same name as an instance variable

**CODE:**

#include<iostream>

using namespace std;

class student\_report

{

public:

char name[30];

int x,y,z;

void student\_data();

void marks();

void final\_data();

};

void student\_report :: student\_data()

{

cout<<"Enter the name::";

cin>>name;

}

void student\_report :: marks()

{

cout<<"\nEnter marks in subject 1:";

cin>>x;

cout<<"\nEnter marks in subject 2:";

cin>>y;

cout<<"\nEnter marks in subject 3:";

cin>>z;

}

void student\_report :: final\_data()

{

this ->student\_data();

this ->marks();

cout<<"\n";

cout<<"\t THE STUDENT REPORT";

cout<<"\nName of student::"<<name;

cout<<"\nMarks in subject 1::"<<x;

cout<<"\nMarks in subject 2::"<<y;

cout<<"\nMarks in subject 3::"<<z;

}

int main()

{

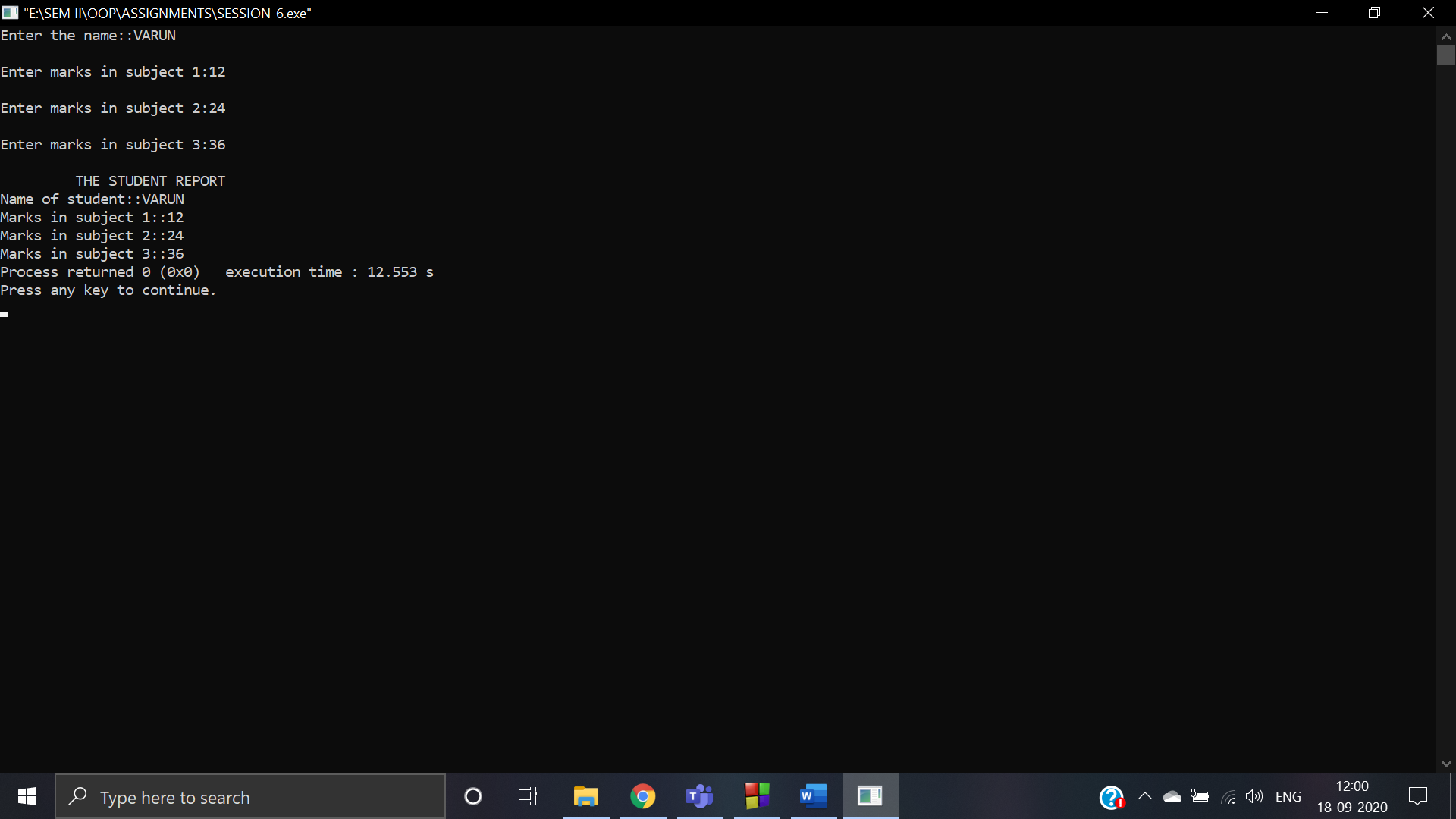
student\_report sr;

sr.final\_data();

return 0;

}

**OUTPUT:**



1. **WAP that can calculate the electricity bill of a person under following conditions:**

|  |  |
| --- | --- |
| **Unit consumed** | **Tariff** |
| **First 100 units** | **Rs. 5 Per unit** |
| **next 200 units** | **Rs. 7 Per unit** |
| **next 300 units** | **Rs. 12 Per unit** |

**The class must include details of the customer like his id, name and the functions read() to accept user inputs, display() to display total bill and a function cal\_tariff() to calculate his total bill. Use appropriate access specifier sections and define all the functions outside the scope of the class using the scope resolution operator.**

**Ans: CODE:**

#include<iostream>

using namespace std;

class electricity\_bill

{

int user\_id,ut\_cd,amt;

public:

char name[12];

void read()

{

cout<<"\nEnter the user id ::";

cin>>user\_id;

cout<<"\nEnter the name of the user::";

cin>>name;

cout<<"\nEnter the units consumed::";

cin>>ut\_cd;

}

void cal\_tariff();

void display();

};

void electricity\_bill :: cal\_tariff()

{

if (ut\_cd <= 100)

{

amt = ut\_cd \* 10;

}

else if (ut\_cd <= 200)

{

amt = (100 \* 10) +

(ut\_cd - 100) \* 15;

}

else if (ut\_cd <= 300)

{

amt = (100 \* 10) +

(100 \* 15) +

(ut\_cd - 200) \* 20;

}

else if (ut\_cd > 300)

{

amt = (100 \* 10) +

(100 \* 15) +

(100 \* 20) +

(ut\_cd - 300) \* 25;

}

}

void electricity\_bill :: display()

{

this -> read();

this -> cal\_tariff();

cout<<"\n";

cout<<"\tThe Electricity bill data";

cout<<"\nUser id::"<<user\_id;

cout<<"\nName of the user::"<<name;

cout<<"\nUnits consumed::"<<ut\_cd;

cout<<"\nElectricity bill::"<<amt;

}

int main()

{

electricity\_bill eb;

eb.display();

return 0;

}

**OUTPUT:**

