Name: ………………………………………………. ID: ………………………………….

	Element	**Description**	**Mark**	**Your mark**
1	Program structure	This will look at the way you designed your program, where did you define your variables, how you are using them, how did you place your functions and the way you are calling them.	20	
2	Function design	This will assess your function design, what kind of parameters you are using/passing, and how did you design the returning values if needed. And how did you document each function.	25	
3	Implementation	How the solution is implemented, how efficient is your solution.	30	
4	Testing	How did you test your program functionality and having a certain testing code for each individual function?	10	
5	Input/ output	How do you perform the data input and how do you present your output.	10	
6	Extras	This represents the extra controls you are adding to have a flexible access and control on the program operations and input/output.	5	
		Total mark	**100%**	

```
#include <iostream>
#include <stdio.h>
#include <string>
#include <limits>
using namespace std;

//Structing
struct Class
{
       string Subject = "";
       string Lecturer= "";
       string RoomName= "";
};
struct TTcell
{
       bool Occupy = false;
       struct Class c;
};

//In-Functions
void Print(TTcell tc[5][9]);
void Clear(TTcell tc[5][9]);
void Insert(TTcell tc[5][9]);
void Show(TTcell tc[5][9]);

int main()
{
       char selection;
       TTcell tc[5][9];

       //Assigning
       struct TTcell Naseer = { true, {"PSP II\t","Dr Naseer","CR10"} };
       struct TTcell Harin = { true, {"Intro 2 OS","Dr Harin","F130"} };
       struct TTcell Thomas = { true, {"Intro 2 Stat", "Dr Thomas","SLT"} };

       //Time Scheduling
       tc[1][0] = Naseer;    tc[1][1] = Naseer;   tc[1][2] = Naseer;   tc[1][3] = Naseer;
tc[3][6] = Naseer;    tc[3][7] = Naseer;   tc[3][8] = Naseer;
       tc[1][5] = Harin;     tc[1][6] = Harin;    tc[2][0] = Harin;    tc[2][1] = Harin;
tc[2][2] = Harin;     tc[4][0] = Harin;    tc[4][1] = Harin;
       tc[2][3] = Thomas;    tc[3][0] = Thomas;   tc[3][1] = Thomas;   tc[3][3] = Thomas;

       //Functioning until Exit
       do
       {
              //Input Asking
              cout << "For Clearing the time table, please type 'C'"
                     <<"\nFor Printing the time table, please type 'P'"
                     << "\nFor Showing specific time table, please type 'S'"
                     << "\nFor Inserting (or Editting) a class, please type 'I'"
                     <<"\nFor Exiting the programme, please type 'E'\n: ";
              cin >> selection;

              //If Print
              if (selection == 'P')
                     Print(tc);
```

```
		//If Clear
		if (selection == 'C')
			Clear(tc);

		//If Insert
		if (selection == 'I')
			Insert(tc);

		//If Show
		if (selection == 'S')
			Show(tc);

	} while (selection !='E');

	cout << "\n\nEND OF PROGRAMME\n\n";

	return 0;
}

void Print(TTcell tc[5][9])
{
	cout << "\n";
	for (int i = 0; i < 5; i++)
	{
		for (int j = 0; j < 9; j++)
		{

			if (tc[i][j].Occupy)
			{
				//Day Output
				if (j == 0)
				{
					if (i == 0) cout << "Monday----------------------\n";
					if (i == 1) cout << "Tuesday---------------------\n";
					if (i == 2) cout << "Wednesday-------------------\n";
					if (i == 3) cout << "Thursday--------------------\n";
					if (i == 4) cout << "Friday----------------------\n";
				}
				//Timetable Output
				cout << j + 9 << ":00\n\t" <<
					tc[i][j].c.Subject << "\t" <<
					tc[i][j].c.Lecturer << "\t" <<
					tc[i][j].c.RoomName << "\t\n";
			}
		}
	}
	cout << "\nCOMPLETED\n\n";
}

void Clear(TTcell tc[5][9])
{
	struct TTcell EmptyCell = { false,{"","",""} };

	for (int i = 0; i < 5; i++)
	{
		for (int j = 0; j < 9; j++)
		{
```

```
                tc[i][j] = EmptyCell;
        }
    }
    cout << "\nCOMPLETED\n\n";
}

void Insert(TTcell tc[5][9])
{
    string stringY;
    int x;
    int y(0);

    //Finding X
    cout << "Time of the class in hour form e.g. 10 : ";
    cin >> x;
    x = x - 9;
    //Finding Y
    cout << "Day of the class in three charaters e.g. Mon : ";
    cin >> stringY;
    if (stringY == "Mon")
        y = 0;
    else if (stringY == "Tue")
        y = 1;
    else if (stringY == "Wed")
        y = 2;
    else if (stringY == "Thu")
        y = 3;
    else if (stringY == "Fri")
        y = 4;

    //Inserting
    cin.ignore(numeric_limits<streamsize>::max(), '\n');
    cout << "Subject : ";
    getline(cin, tc[y][x].c.Subject);
    cout << "Lecturer : ";
    getline(cin, tc[y][x].c.Lecturer);
    cout << "Room Name : ";
    getline(cin, tc[y][x].c.RoomName);

    tc[y][x].Occupy = true;

    cout << "\nCOMPLETED\n\n";
}

void Show(TTcell tc[5][9])
{
    int x;
    int y;
    string stringY;

    //Finding X
    cout << "Time of the class in hour form e.g. 10 : ";
    cin >> x;
    x = x - 9;
    //Finding Y
    cout << "Day of the class in three charaters e.g. Mon : ";
    cin >> stringY;
    if (stringY == "Mon")
        y = 0;
    else if (stringY == "Tue")
        y = 1;
    else if (stringY == "Wed")
        y = 2;
    else if (stringY == "Thu")
        y = 3;
    else if (stringY == "Fri")
        y = 4;

    if (tc[y][x].Occupy)
        {
        //Timetable Output
            cout << "\n"<<stringY<<"day----------------------\n"<<
            x + 9 << ":00\n\t" <<
            tc[y][x].c.Subject << "\t" <<
            tc[y][x].c.Lecturer << "\t" <<
            tc[y][x].c.RoomName << "\t\n";
        }
    else
    {
        cout << "\n" << stringY << "day----------------------\n" <<
            "NO CLASS\n";
    }
    cout << "\nCOMPLETED\n\n";
}
```

```
#include <stdio.h>
#include <iostream>
#include <string>
#include <numeric>

using namespace std;

//Structing Records
struct CarRecord
{
       string CarMake;
       string CarModel;
       float  EngineSize = 0;
       string Colour;
       int           Mileage = 0;
       int           Price = 0;
};

//Functions
//void ReadCar(CarRecord c[5]);

int main()
{
       struct CarRecord c[5];

       c[0] = { "Lexus","CT200h",1.8,"Blue",22500,9355 };
       c[1] = { "Volvo","S80",2.4,"Silver",80345,8990 };
       c[2] = { "Audi","A3",2,"Green",51800,12995 };
       c[3] = { "Honda","Jazz",1.4,"Silver",30400,6890 };
       c[4] = { "BMW","2Series",1.5,"Gray",29433,14500 };
       c[5] = { "Nissan","Micra",1,"Black",80700,5995 };

       cout << "\n\nEND OF PROGRAM \n\n";
}

/*
void ReadCar(struct CarRecord c[5])
{
       c[0] = { "Lexus","CT200h",1.8,"Blue",22500,9355 };
       c[1] = { "Volvo","S80",2.4,"Silver",80345,8990 };
       c[2] = { "Audi","A3",2,"Green",51800,12995 };
       c[3] = { "Honda","Jazz",1.4,"Silver",30400,6890 };
       c[4] = { "BMW","2Series",1.5,"Gray",29433,14500 };
       c[5] = { "Nissan","Micra",1,"Black",80700,5995 };
}*/
```