Midterm Skills Exam	
Set A	
Course Code: CPE007	**Program:** Computer Engineering
Course Title: Programming Logic and Design	**Date Performed:10/9/25**
Section:CPE11S1	**Date Submitted:10/9/25**
Name(s): Will Stuart D. Ponce Jr.	**Instructor:** Engr. Jimlord M. Quejado

6.Output:

1.Problem Description:

Create a program that records the grades of several students and computes their average grades. Use structures to store each student's data, arrays to handle multiple students, and loops for data input and processing.

Code:

```
#include <iostream>
#include <iomanip>
#include <string>
using namespace std;

const int MAX_STUDENTS = 10;
const int NUM_GRADES = 3;

struct Student {
    string name;
    string studentNumber;
    int grades[NUM_GRADES];
    float average;
};
int main() {
    int numStudents;
    cout << "Enter number of students (max 10): ";
    cin >> numStudents;
    cin.ignore();
    if (numStudents > MAX_STUDENTS || numStudents <= 0) {
        cout << "Invalid number of students. Please enter between 1 and 10." << endl;
        return 1;
    }
    Student students[MAX_STUDENTS];
    int topIndex = 0;
    for (int i = 0; i < numStudents; ++i) {
        cout << "\nEnter details for student " << i + 1 << ":\n";

        cout << "Name: ";
        getline(cin, students[i].name);

        cout << "Student Number: ";
        getline(cin, students[i].studentNumber);

        cout << "Enter 3 grades:\n";
        int sum = 0;
        for (int j = 0; j < NUM_GRADES; ++j) {
            cin >> students[i].grades[j];
            sum += students[i].grades[j];
        }
        cin.ignore();

        students[i].average = static_cast<float>(sum) / NUM_GRADES;


        if (students[i].average > students[topIndex].average) {
            topIndex = i;
        }
    }
    cout << "\n--------------------------------------------------------------------------\n";
    cout << "                STUDENT GRADE REPORT\n";
    cout << "--------------------------------------------------------------------------\n";
    cout << left << setw(20) << "Name"
         << setw(15) << "Student No."
         << setw(20) << "Grades"
         << "Average\n";
    cout << "--------------------------------------------------------------------------\n";
    for (int i = 0; i < numStudents; ++i) {
        cout << left << setw(20) << students[i].name
             << setw(15) << students[i].studentNumber;

        for (int j = 0; j < NUM_GRADES; ++j) {
            cout << students[i].grades[j] << "  ";
        }

        cout << setw(10) << fixed << setprecision(2) << students[i].average << endl;
    }
    cout << "--------------------------------------------------------------------------\n";
    cout << "Top Student: " << students[topIndex].name
         << " (Average: " << fixed << setprecision(2) << students[topIndex].average << ")\n";
    cout << "--------------------------------------------------\n";

    return 0;
}
```

```
Name: kyle
Student Number: 21000003
Enter 3 grades:
90
85
97

Enter details for student 2:
Name: Paul
Student Number: 2211412412
Enter 3 grades:
78
77
76

------------------------------------------------------------------------
                STUDENT GRADE REPORT
------------------------------------------------------------------------
Name                Student No.    Grades              Average
------------------------------------------------------------------------
kyle                21000003       90  85  97  90.67
Paul                2211412412     78  77  76  77.00
------------------------------------------------------------------------
Top Student: kyle (Average: 90.67)
------------------------------------------------

--------------------------------
Process exited after 83.66 seconds with return value 0
Press any key to continue . . .
```

7. Supplementary Activity:

8. Conclusion:While it might be difficult to fully comprehend this code at the beginning, it is critical to allow it to settle in your mind and digest it. The first step should always be to read it and try to comprehend the code, then try to build your code and your approach to the problem.

This type of code can be beneficial in the real world. For example, in game design, you will be the game designer. You can also be used in the automation of student grade calculations. Instead of entering twenty student grades manually, you could create a C++ program that enters the grades for you.

There are so many opportunities with the essentials of C++. Programming is not just limited to tools and games, but it is used throughout systems and much more in today's world. If you enjoy coding, typing, and computing, programming can be used in a stress releasing way! While if you don't enjoy coding, coding is not for everyone.