OOP Assignment (Chapter 6)

Exercise 6.4:

Code:

exercise4.cpp

```cpp
#include <iostream>
using namespace std;

class Employee
{
private:
    int EmpNum;
    float Compensation;

public:
    void GetInfo()
    {
        cout << "Enter Employee Number: ";
        cin >> EmpNum;
        cout << "Enter Compensation: ";
        cin >> Compensation;
    }
    void DisplayInfo()
    {
        cout << "Employee Number: " << EmpNum << endl;
        cout << "Compensation: " << Compensation << endl;
    }
};
int main()
{
    Employee emp[3]; // Array of 3 employees

    cout << "Enter details for 3 employees:\n";
    for (int i = 0; i < 3; i++)
    {
        cout << "\nEmployee " << i + 1 << ":\n";
        emp[i].GetInfo();
    }
    cout << "\n--- Employee Details ---\n";
    for (int i = 0; i < 3; i++)
    {
        cout << "\nEmployee " << i + 1 << ":\n";
        emp[i].DisplayInfo();
    }
    return 0;
}
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Enter details for 3 employees:

Employee 1:
Enter Employee Number: 1
Enter Compensation: 50000

Employee 2:
Enter Employee Number: 2
Enter Compensation: 3000

Employee 3:
Enter Employee Number: 3
Enter Compensation: 498000

--- Employee Details ---

Employee 1:
Employee Number: 1
Compensation: 50000

Employee 2:
Employee Number: 2
Compensation: 3000

Employee 3:
Employee Number: 3
Compensation: 498000
```

Exercise 6.5:

Code:

exercise4.cpp | exercise5.cpp

exercise5.cpp > main()

```cpp
#include <iostream>
using namespace std;

class Date{
private:
    int day;
    int month;
    int year;
public:
    void GetDate() {
        char slash;
        cout << "Enter date (DD/MM/YYYY): ";
        cin >> day >> slash >> month >> slash >> year;
    }
    void displayDate() {
        cout << "Date: " << day << "/" << month << "/" << year << endl;
    }
};

int main(){

    Date d;
    d.GetDate();
    cout << "Ther entered date is: ";
    d.displayDate();

    return 0;
}
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS                                Code

Active code page: 65001

C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>cd "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\" && g++ exercise5.cp
p -o exercise5 && "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\"exercise5
Enter date (DD/MM/YYYY): 12/12/2025
Ther entered date is: Date: 12/12/2025

C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>
```

Exercise 6.6:

Code:

exercise4.cpp | exercise5.cpp | exercise6.cpp

exercise6.cpp > Date > showDate() const

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

// Date class
class Date {
private:
    int day, month, year;

public:
    void getDate() {
        cout << "Enter hire date (dd mm yyyy): ";
        cin >> day >> month >> year;
    }

    void showDate() const {
        cout << day << "/" << month << "/" << year;
    }
};

// Enum for employee type
enum etype { laborer, secretary, manager, accountant, executive, researcher };

class Employee {
private:
    int EmpNum;
    float Compensation;
    Date hireDate;   // object of Date class
    etype EmpType;   // enum variable

public:
    void getemploy() {
        cout << "Enter Employee Number: ";
        cin >> EmpNum;
        cout << "Enter Compensation: ";
        cin >> Compensation;
```

```
        hireDate.getDate();

        char type;
        cout << "Enter employee type (l=Laborer, s=Secretary, m=Manager, "
             << "a=Accountant, e=Executive, r=Researcher): ";
        cin >> type;

        switch(type) {
            case 'l': EmpType = laborer; break;
            case 's': EmpType = secretary; break;
            case 'm': EmpType = manager; break;
            case 'a': EmpType = accountant; break;
            case 'e': EmpType = executive; break;
            case 'r': EmpType = researcher; break;
            default: cout << "Invalid type! Defaulting to laborer.\n";
                     EmpType = laborer;
        }
    }

    void putemploy() const {
        cout << "\nEmployee Number: " << EmpNum
             << "\nCompensation: " << Compensation
             << "\nHire Date: ";
        hireDate.showDate();

        cout << "\nEmployee Type: ";
        switch(EmpType) {
            case laborer:   cout << "Laborer"; break;
            case secretary: cout << "Secretary"; break;
            case manager:   cout << "Manager"; break;
            case accountant:cout << "Accountant"; break;
            case executive: cout << "Executive"; break;
            case researcher:cout << "Researcher"; break;
```

```
        }
        cout << endl;
    }
};

int main() {
    Employee emp[3];

    cout << "Enter details for 3 employees:\n";
    for (int i = 0; i < 3; i++) {
        cout << "\nEmployee " << i + 1 << ":\n";
        emp[i].getemploy();
    }

    cout << "\n--- Employee Details ---\n";
    for (int i = 0; i < 3; i++) {
        cout << "\nEmployee " << i + 1 << ":\n";
        emp[i].putemploy();
    }

    return 0;
}
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Active code page: 65001

C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>cd "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\" && g++ exercise6.cp
p -o exercise6 && "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\"exercise6
Enter details for 3 employees:

Employee 1:
Enter Employee Number: 1
Enter Compensation: 1200
Enter hire date (dd mm yyyy): 12 12 2025
Enter employee type (l=Laborer, s=Secretary, m=Manager, a=Accountant, e=Executive, r=Researcher): s

Employee 2:
Enter Employee Number: 2
Enter Compensation: 1500
Enter hire date (dd mm yyyy): 12 11 2021
Enter employee type (l=Laborer, s=Secretary, m=Manager, a=Accountant, e=Executive, r=Researcher): m

Employee 3:
Enter Employee Number: 1
Enter Compensation: 50000
Enter hire date (dd mm yyyy): 13 4 2019
Enter employee type (l=Laborer, s=Secretary, m=Manager, a=Accountant, e=Executive, r=Researcher): r

--- Employee Details ---

Employee 1:

Employee Number: 1
Compensation: 1200
Hire Date: 12/12/2025
Employee Type: Secretary

Employee 2:

Employee Number: 2
Compensation: 1500
Hire Date: 12/11/2021
Employee Type: Manager
```

Exercise 6.7:

Code:

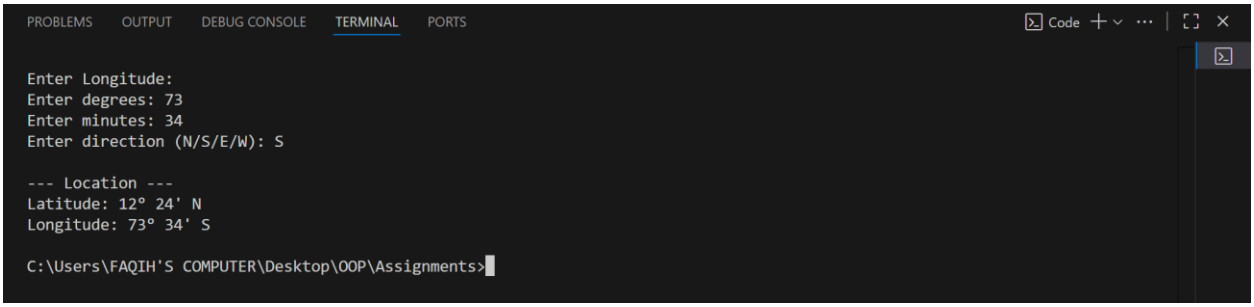
exercise4.cpp | exercise5.cpp | exercise6.cpp | exercise7.cpp

exercise7.cpp > Angle

```cpp
#include <iostream>
using namespace std;

class Angle {
private:
    int degrees;
    float minutes;
    char direction;

public:
    Angle() : degrees(0), minutes(0.0), direction('E') {}  // default constructor

    void getAngle() {
        cout << "Enter degrees: ";
        cin >> degrees;
        cout << "Enter minutes: ";
        cin >> minutes;
        cout << "Enter direction (N/S/E/W): ";
        cin >> direction;
    }

    void displayAngle() const {
        cout << degrees << "° " << minutes << "' " << direction;
    }
};

```

```cpp

int main() {
    Angle latitude, longitude;

    cout << "Enter Latitude:\n";
    latitude.getAngle();

    cout << "\nEnter Longitude:\n";
    longitude.getAngle();

    cout << "\n--- Location ---\n";
    cout << "Latitude: ";
    latitude.displayAngle();
    cout << "\nLongitude: ";
    longitude.displayAngle();
    cout << endl;

    return 0;
}

```

Output:



Exercise 6.8:

Code:

exercise4.cpp exercise5.cpp exercise6.cpp exercise7.cpp exercise8.cpp

exercise8.cpp > ...

```cpp
#include <iostream>
using namespace std;

class Serial {
private:
    static int count;
    int serialNum;

public:
    Serial() {
        count++;
        serialNum = count;
    }

    void report() {
        cout << "I am object number " << serialNum << endl;
    }
};
int Serial::count = 0;

int main() {
    Serial obj1, obj2, obj3;

    obj1.report();
    obj2.report();
    obj3.report();

    return 0;
}
```

Output:

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Code

```
Active code page: 65001

C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>cd "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\" && g++ exercise8.cp
p -o exercise8 && "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\"exercise8
I am object number 1
I am object number 2
I am object number 3

C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>
```

Exercise 6.9:

Code:

exercise4.cpp exercise5.cpp exercise6.cpp exercise7.cpp exercise8.cpp exercise9.cpp

exercise9.cpp > ...

```cpp
#include <iostream>
using namespace std;

class Fraction
{
private:
    int num, den;

public:
    void getFraction()
    {
        char slash;
        cout << "Enter fraction (a/b): ";
        cin >> num >> slash >> den;
    }
    void displayFraction() const
    {
        cout << num << "/" << den;
    }
    Fraction add(Fraction f)
    {
        Fraction temp;
        temp.num = num * f.den + f.num * den;
        temp.den = den * f.den;
        return temp;
    }
};
```

```cpp

int main()
{
    char choice;
    do
    {
        Fraction f1, f2, sum;

        f1.getFraction();
        f2.getFraction();

        sum = f1.add(f2);

        cout << "Sum = ";
        sum.displayFraction();
        cout << endl;

        cout << "Do you want to continue? (y/n): ";
        cin >> choice;

    } while (choice == 'y' || choice == 'Y');

    return 0;
}

```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS                                   Code

Active code page: 65001

C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>cd "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\" && g++ exercise9.cp
p -o exercise9 && "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\"exercise9
Enter fraction (a/b): 1/2
Enter fraction (a/b): 3/4
Sum = 10/8
Do you want to continue? (y/n): y
Enter fraction (a/b): 5/9
Enter fraction (a/b): 2/7
Sum = 53/63
Do you want to continue? (y/n): n

C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>
```

Ln 28, Col 1 Spaces: 4 UTF-8 LF {} C++ Win32 Prettier

Exercise 6.10:

Code:

exercise4.cpp | exercise5.cpp | exercise6.cpp | exercise7.cpp | exercise8.cpp | exercise9.cpp | exercise10.cpp

exercise10.cpp > ...

```cpp
#include <iostream>
using namespace std;

class Angle {
private:
    int degrees;
    float minutes;
    char direction;   // N, S, E, W

public:
    void getAngle() {
        cout << "Enter degrees: ";
        cin >> degrees;
        cout << "Enter minutes: ";
        cin >> minutes;
        cout << "Enter direction (N/S/E/W): ";
        cin >> direction;
    }

    void displayAngle() const {
        cout << degrees << "° " << minutes << "' " << direction;
    }
};

class Ship {
private:
    static int count;
    int serialNum;
    Angle latitude;
    Angle longitude;

public:
    Ship() {
        count++;
        serialNum = count;
    }
```

```
    void getPosition() {
        cout << "Enter Latitude:\n";
        latitude.getAngle();
        cout << "Enter Longitude:\n";
        longitude.getAngle();
    }

    void report() const {
        cout << "\nShip Serial Number: " << serialNum << endl;
        cout << "Latitude: ";
        latitude.displayAngle();
        cout << "\nLongitude: ";
        longitude.displayAngle();
        cout << endl;
    }
};

int Ship::count = 0; // initialize static count

int main() {
    Ship ships[3];

    for(int i = 0; i < 3; i++) {
        cout << "\nEnter details for Ship " << i+1 << ":\n";
        ships[i].getPosition();
    }

    cout << "\n--- Ship Details ---\n";
    for(int i = 0; i < 3; i++) {
        ships[i].report();
    }

    return 0;
```

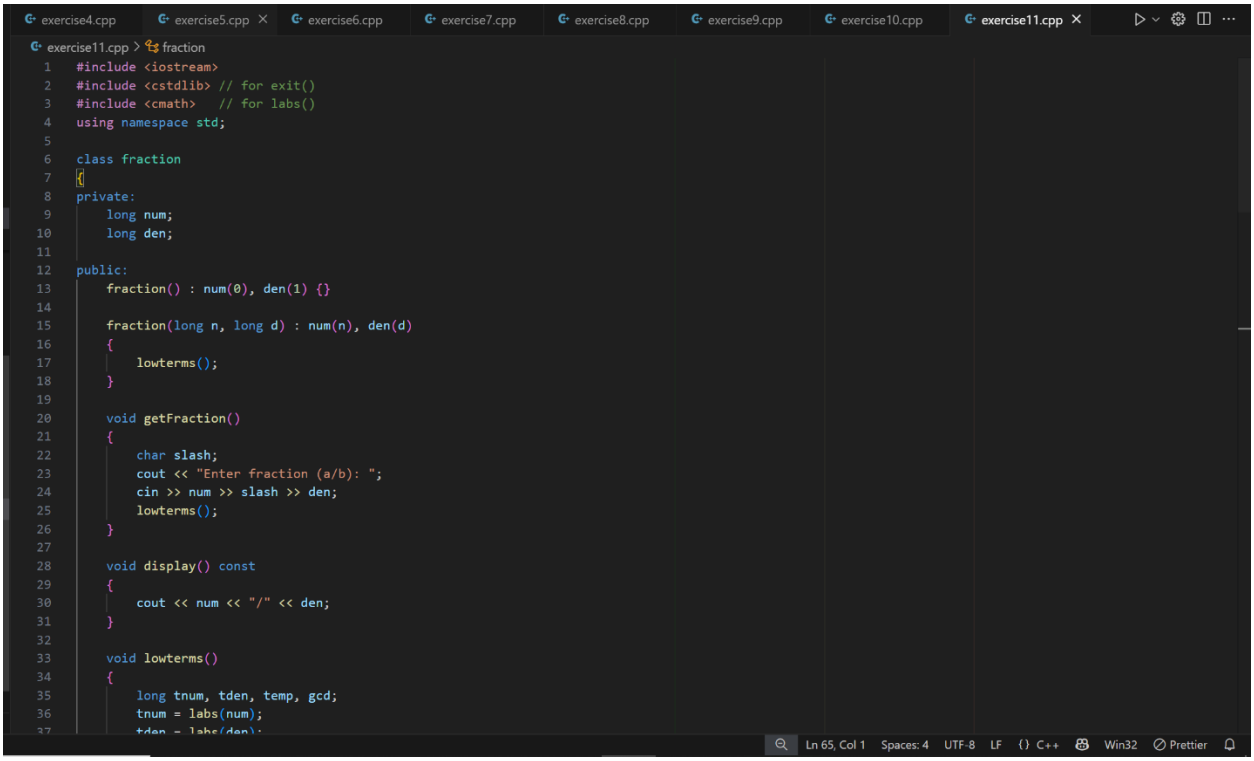
Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS                                   Code

Active code page: 65001

C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>cd "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\" && g++ exercise10.c
pp -o exercise10 && "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\"exercise10

Enter details for Ship 1:
Enter Latitude:
Enter degrees: 12
Enter minutes: 23
Enter direction (N/S/E/W): S
Enter Longitude:
Enter degrees: 98
Enter minutes: 89
Enter direction (N/S/E/W): W

Enter details for Ship 2:
Enter Latitude:
Enter degrees: 12
Enter minutes: 32
Enter direction (N/S/E/W): E
Enter Longitude:
Enter degrees: 98
Enter minutes: 98
Enter direction (N/S/E/W): N

Enter details for Ship 3:
Enter Latitude:
Enter degrees: 23
Enter minutes: 32
Enter direction (N/S/E/W): E
Enter Longitude:
Enter degrees: 78
Enter minutes: 98
Enter direction (N/S/E/W): E
```

```
--- Ship Details ---

Ship Serial Number: 1
Latitude: 12° 23' S
Longitude: 98° 89' W

Ship Serial Number: 2
Latitude: 12° 32' E
Longitude: 98° 98' N

Ship Serial Number: 3
Latitude: 23° 32' E
Longitude: 78° 98' E
```

Exercise 6.11:

Code:



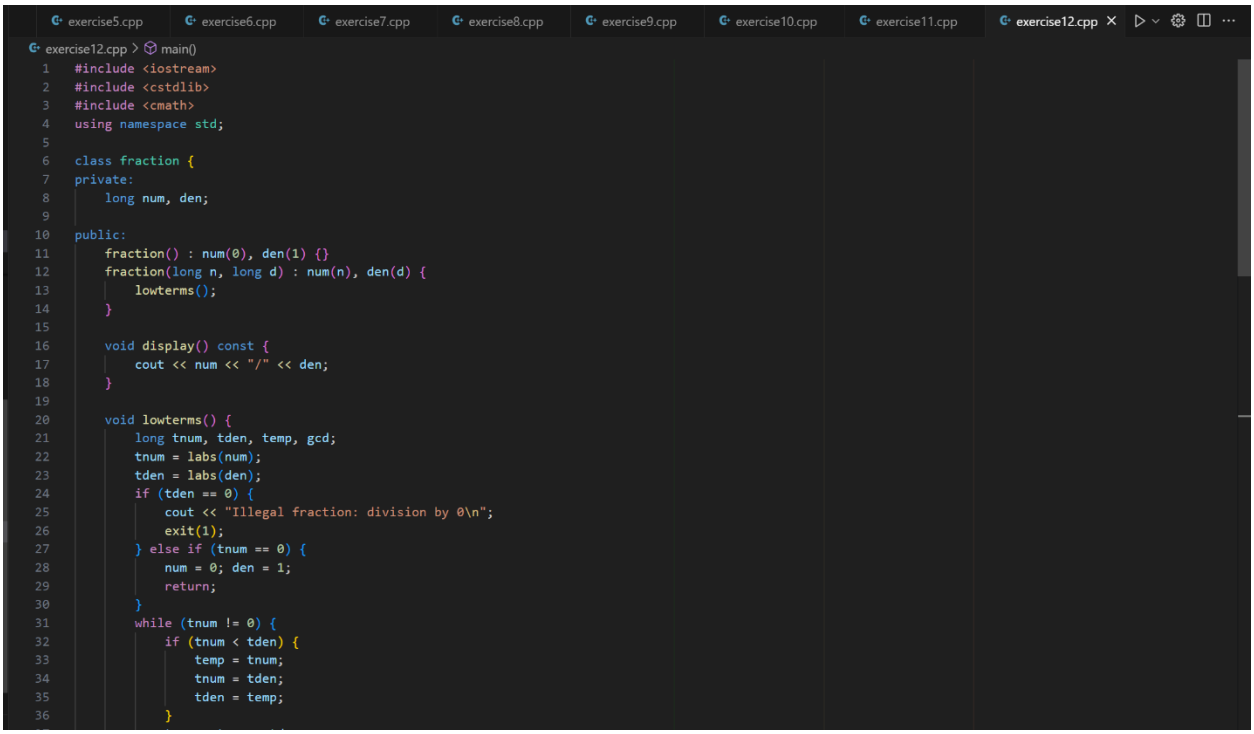
```
        tden = labs(den);
        if (tden == 0)
        {
            cout << "Illegal fraction: division by 0\n";
            exit(1);
        }
        else if (tnum == 0)
        { // check for 0/n
            num = 0;
            den = 1;
            return;
        }

        while (tnum != 0)
        {
            if (tnum < tden)
            {
                temp = tnum;
                tnum = tden;
                tden = temp;
            }
            tnum = tnum - tden;
        }
        gcd = tden;

        num = num / gcd;
        den = den / gcd;
    }

    fraction operator+(const fraction &f) const
    {
        return fraction(num * f.den + den * f.num, den * f.den);
    }

    fraction operator-(const fraction &f) const
    {
        return fraction(num * f.den - den * f.num, den * f.den);
    }

    fraction operator*(const fraction &f) const
    {
        return fraction(num * f.num, den * f.den);
    }

    fraction operator/(const fraction &f) const
    {
        return fraction(num * f.den, den * f.num);
    }
};

int main()
{
    fraction f1, f2, result;
    char op;

    cout << "Enter first fraction: ";
    f1.getFraction();
    cout << "Enter operator (+, -, *, /): ";
    cin >> op;
    cout << "Enter second fraction: ";
    f2.getFraction();

    switch (op)
    {
    case '+':
        result = f1 + f2;
        break;
```

```
            result = f1 + f2;
            break;
        case '-':
            result = f1 - f2;
            break;
        case '*':
            result = f1 * f2;
            break;
        case '/':
            result = f1 / f2;
            break;
        default:
            cout << "Unknown operator\n";
            exit(1);
        }

        cout << "Result = ";
        result.display();
        cout << endl;

        return 0;
    }

```

Output:

Exercise 6.12:

Code:



```cpp
#include <iostream>
#include <cstdlib>
#include <cmath>
using namespace std;

class fraction {
private:
    long num, den;

public:
    fraction() : num(0), den(1) {}
    fraction(long n, long d) : num(n), den(d) {
        lowterms();
    }

    void display() const {
        cout << num << "/" << den;
    }

    void lowterms() {
        long tnum, tden, temp, gcd;
        tnum = labs(num);
        tden = labs(den);
        if (tden == 0) {
            cout << "Illegal fraction: division by 0\n";
            exit(1);
        } else if (tnum == 0) {
            num = 0; den = 1;
            return;
        }
        while (tnum != 0) {
            if (tnum < tden) {
                temp = tnum;
                tnum = tden;
                tden = temp;
            }
```

```cpp
            tnum = tnum - tden;
        }
        gcd = tden;
        num /= gcd;
        den /= gcd;
    }

    fraction operator * (const fraction& f) const {
        return fraction(num * f.num, den * f.den);
    }};
int main() {
    int denom;
    cout << "Enter a denominator: ";
    cin >> denom;
    cout << "\t";
    for (int i = 1; i < denom; i++) {
        fraction f(i, denom);
        f.display();
        cout << "\t";}
    cout << endl;
    for (int i = 1; i < denom; i++) {
        fraction rowFrac(i, denom);
        rowFrac.display();
        cout << "\t";
        for (int j = 1; j < denom; j++) {
            fraction colFrac(j, denom);
            fraction result = rowFrac * colFrac;
            result.display();
            cout << "\t";
        }
        cout << endl;}
    return 0;
}
```

Output:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Code
Active code page: 65001
C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>cd "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\" && g++ exercise12.cpp -o exercise12 && "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\"exercise12
Enter a denominator: 4
1/4 1/2 3/4
1/4 1/16 1/8 3/16
1/2 1/8 1/4 3/8
3/4 3/16 3/8 9/16
C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>