

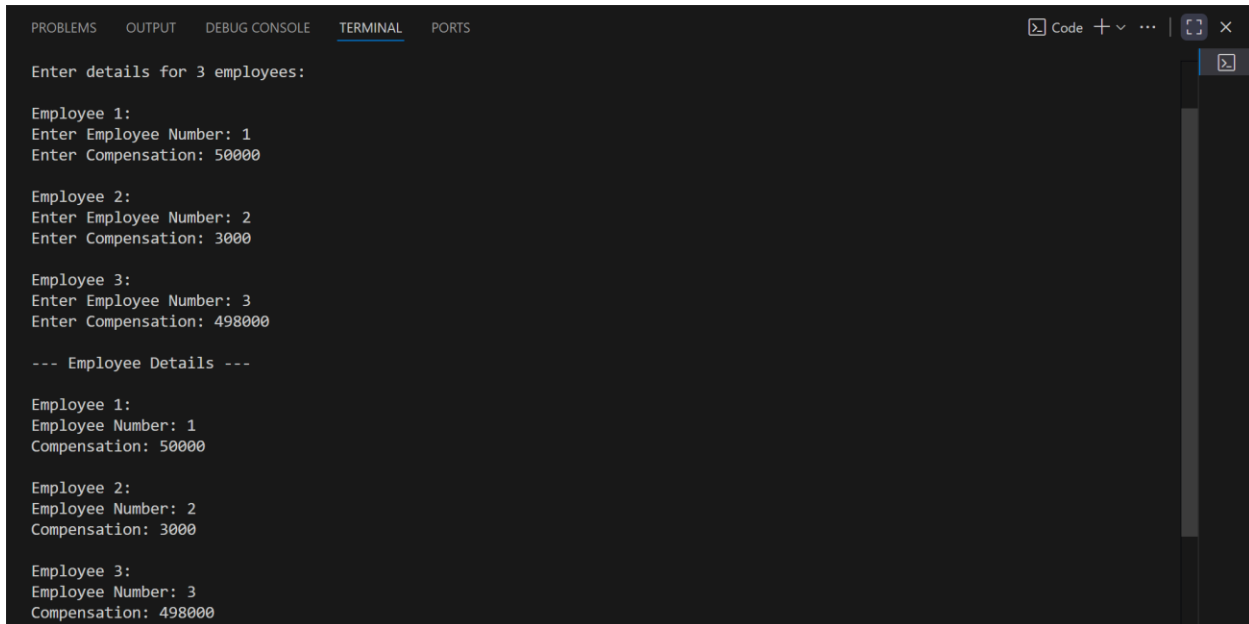
## OOP Assignment (Chapter 6)

### **Exercise 6.4:**

#### **Code:**

```
exercise4.cpp X
exercise4.cpp > ...
1  #include <iostream>
2  using namespace std;
3
4  class Employee
5  {
6  private:
7      int EmpNum;
8      float Compensation;
9
10 public:
11     void GetInfo()
12     {
13         cout << "Enter Employee Number: ";
14         cin >> EmpNum;
15         cout << "Enter Compensation: ";
16         cin >> Compensation;
17     }
18     void DisplayInfo()
19     {
20         cout << "Employee Number: " << EmpNum << endl;
21         cout << "Compensation: " << Compensation << endl;
22     }
23 };
24 int main()
25 {
26     Employee emp[3]; // Array of 3 employees
27
28     cout << "Enter details for 3 employees:\n";
29     for (int i = 0; i < 3; i++)
30     {
31         cout << "\nEmployee " << i + 1 << ":\n";
32         emp[i].GetInfo();
33     }
34     cout << "\n--- Employee Details ---\n";
35     for (int i = 0; i < 3; i++)
36     {
37         cout << "\nEmployee " << i + 1 << ":\n";
38         emp[i].DisplayInfo();
39     }
40     return 0;
41 }
42
```

#### **Output:**



```
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 ---

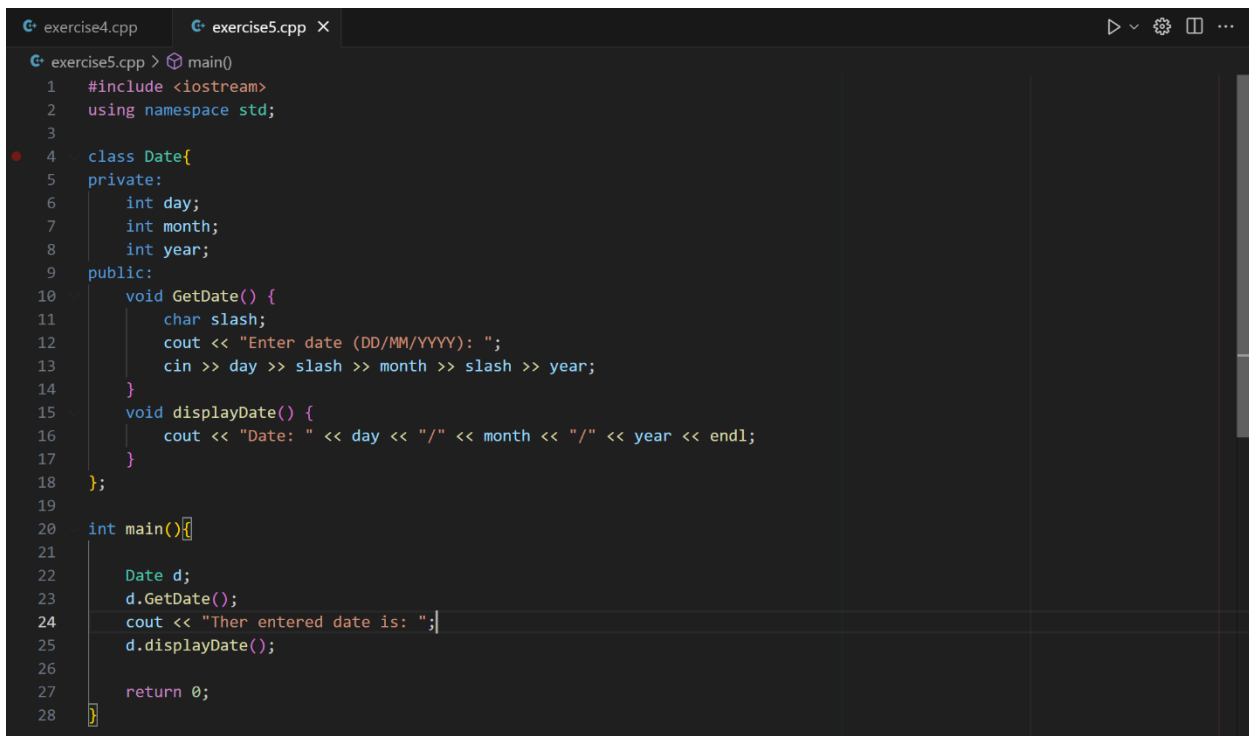
Employee 1:
Employee Number: 1
Compensation: 50000

Employee 2:
Employee Number: 2
Compensation: 3000

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

## Exercise 6.5:

### Code:



```
exercise4.cpp  exercise5.cpp X
exercise5.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  class Date{
5  private:
6      int day;
7      int month;
8      int year;
9  public:
10     void GetDate() {
11         char slash;
12         cout << "Enter date (DD/MM/YYYY): ";
13         cin >> day >> slash >> month >> slash >> year;
14     }
15     void displayDate() {
16         cout << "Date: " << day << "/" << month << "/" << year << endl;
17     }
18 };
19
20 int main(){
21
22     Date d;
23     d.GetDate();
24     cout << "The entered date is: ";
25     d.displayDate();
26
27     return 0;
28 }
```

### 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++ exercise5.cpp -o exercise5 && "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\"exercise5
Enter date (DD/MM/YYYY): 12/12/2025
The 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 X
exercise6.cpp > Date > showDate() const
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  // Date class
6  class Date {
7  private:
8      int day, month, year;
9
10 public:
11     void getDate() {
12         cout << "Enter hire date (dd mm yyyy): ";
13         cin >> day >> month >> year;
14     }
15
16     void showDate() const {
17         cout << day << "/" << month << "/" << year;
18     }
19 };
20
21 // Enum for employee type
22 enum etype { laborer, secretary, manager, accountant, executive, researcher };
23
24 class Employee {
25 private:
26     int EmpNum;
27     float Compensation;
28     Date hireDate; // object of Date class
29     etype EmpType; // enum variable
30
31 public:
32     void getemploy() {
33         cout << "Enter Employee Number: ";
34         cin >> EmpNum;
35         cout << "Enter Compensation: ";
36         cin >> Compensation;
```

```

37     hireDate.getDate();
38
39
40     char type;
41     cout << "Enter employee type (l=Laborer, s=Secretary, m=Manager, "
42           "a=Accountant, e=Executive, r=Researcher): ";
43     cin >> type;
44
45     switch(type) {
46     case 'l': EmpType = laborer; break;
47     case 's': EmpType = secretary; break;
48     case 'm': EmpType = manager; break;
49     case 'a': EmpType = accountant; break;
50     case 'e': EmpType = executive; break;
51     case 'r': EmpType = researcher; break;
52     default: cout << "Invalid type! Defaulting to laborer.\n";
53             EmpType = laborer;
54     }
55 }
56
57 void putemploy() const {
58     cout << "\nEmployee Number: " << EmpNum
59           << "\nCompensation: " << Compensation
60           << "\nHire Date: ";
61     hireDate.showDate();
62
63     cout << "\nEmployee Type: ";
64     switch(EmpType) {
65     case laborer: cout << "Laborer"; break;
66     case secretary: cout << "Secretary"; break;
67     case manager: cout << "Manager"; break;
68     case accountant: cout << "Accountant"; break;
69     case executive: cout << "Executive"; break;
70     case researcher: cout << "Researcher"; break;
71     }
72     cout << endl;
73 }
74 };
75
76 int main() {
77     Employee emp[3];
78
79     cout << "Enter details for 3 employees:\n";
80     for (int i = 0; i < 3; i++) {
81         cout << "\nEmployee " << i + 1 << ": \n";
82         emp[i].getemploy();
83     }
84
85     cout << "\n--- Employee Details ---\n";
86     for (int i = 0; i < 3; i++) {
87         cout << "\nEmployee " << i + 1 << ": \n";
88         emp[i].putemploy();
89     }
90
91     return 0;
92 }
93

```

**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:

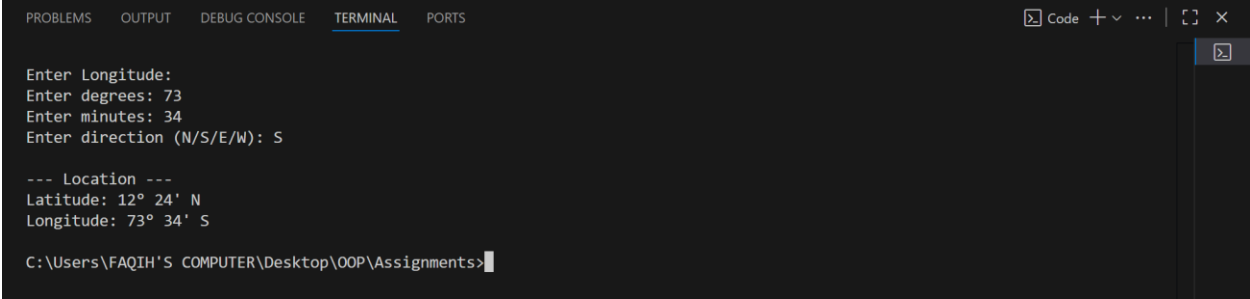
```

exercise4.cpp  exercise5.cpp  exercise6.cpp  exercise7.cpp X
exercise7.cpp > Angle
1  #include <iostream>
2  using namespace std;
3
4  class Angle {
5  private:
6      int degrees;
7      float minutes;
8      char direction;
9
10 public:
11     Angle() : degrees(0), minutes(0.0), direction('E') {} // default constructor
12
13     void getAngle() {
14         cout << "Enter degrees: ";
15         cin >> degrees;
16         cout << "Enter minutes: ";
17         cin >> minutes;
18         cout << "Enter direction (N/S/E/W): ";
19         cin >> direction;
20     }
21
22     void displayAngle() const {
23         cout << degrees << "° " << minutes << "' " << direction;
24     }
25 };
26

```

```
26
27 int main() {
28     Angle latitude, longitude;
29
30     cout << "Enter Latitude:\n";
31     latitude.getAngle();
32
33     cout << "\nEnter Longitude:\n";
34     longitude.getAngle();
35
36     cout << "\n--- Location ---\n";
37     cout << "Latitude: ";
38     latitude.displayAngle();
39     cout << "\nLongitude: ";
40     longitude.displayAngle();
41     cout << endl;
42
43     return 0;
44 }
45
```

## Output:



The screenshot shows a terminal window with the following output:

```
Enter Longitude:
Enter degrees: 73
Enter minutes: 34
Enter direction (N/S/E/W): S

--- Location ---
Latitude: 12° 24' N
Longitude: 73° 34' S

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

The terminal window has tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (selected), and PORTS. The path at the bottom is C:\Users\FAQIH\S COMPUTER\Desktop\OOP\Assignments>.

## Exercise 6.8:

### Code:

```
exercise4.cpp  exercise5.cpp  exercise6.cpp  exercise7.cpp  exercise8.cpp X
exercise8.cpp > ...
1  #include <iostream>
2  using namespace std;
3
4  class Serial {
5  private:
6      static int count;
7      int serialNum;
8
9  public:
10     Serial() {
11         count++;
12         serialNum = count;
13     }
14
15     void report() {
16         cout << "I am object number " << serialNum << endl;
17     }
18 };
19 int Serial::count = 0;
20
21 int main() {
22     Serial obj1, obj2, obj3;
23
24     obj1.report();
25     obj2.report();
26     obj3.report();
27
28     return 0;
29 }
30
```

## 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++ 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:

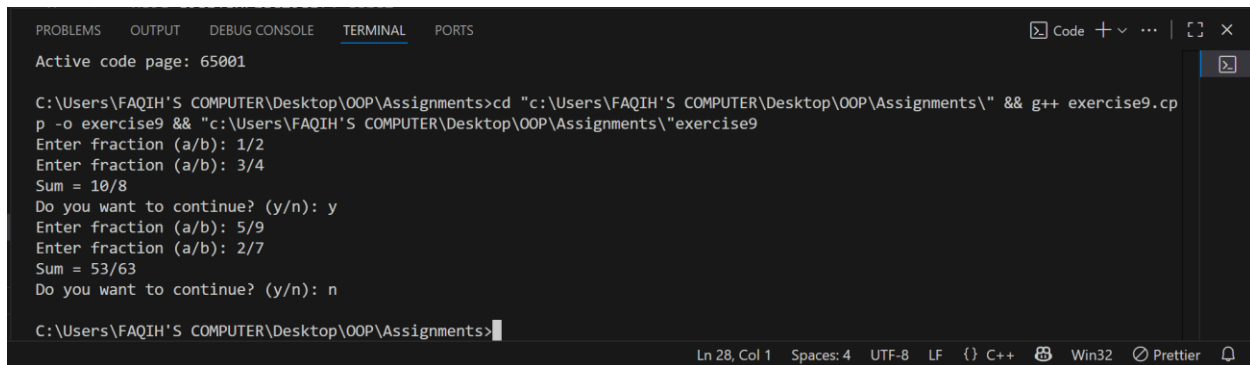
### Code:

```
exercise4.cpp  exercise5.cpp  exercise6.cpp  exercise7.cpp  exercise8.cpp  exercise9.cpp X
exercise9.cpp > ...
1  #include <iostream>
2  using namespace std;
3
4  class Fraction
5  {
6  private:
7      int num, den;
8
9  public:
10     void getFraction()
11     {
12         char slash;
13         cout << "Enter fraction (a/b): ";
14         cin >> num >> slash >> den;
15     }
16     void displayFraction() const
17     {
18         cout << num << "/" << den;
19     }
20     Fraction add(Fraction f)
21     {
22         Fraction temp;
23         temp.num = num * f.den + f.num * den;
24         temp.den = den * f.den;
25         return temp;
26     }
27 };

28
29 int main()
30 {
31     char choice;
32     do
33     {
34         Fraction f1, f2, sum;
35
36         f1.getFraction();
37         f2.getFraction();
38
39         sum = f1.add(f2);
40
41         cout << "Sum = ";
42         sum.displayFraction();
43         cout << endl;
44
45         cout << "Do you want to continue? (y/n): ";
46         cin >> choice;
47     } while (choice == 'y' || choice == 'Y');
48
49     return 0;
50 }
51
52
```

**Output:**





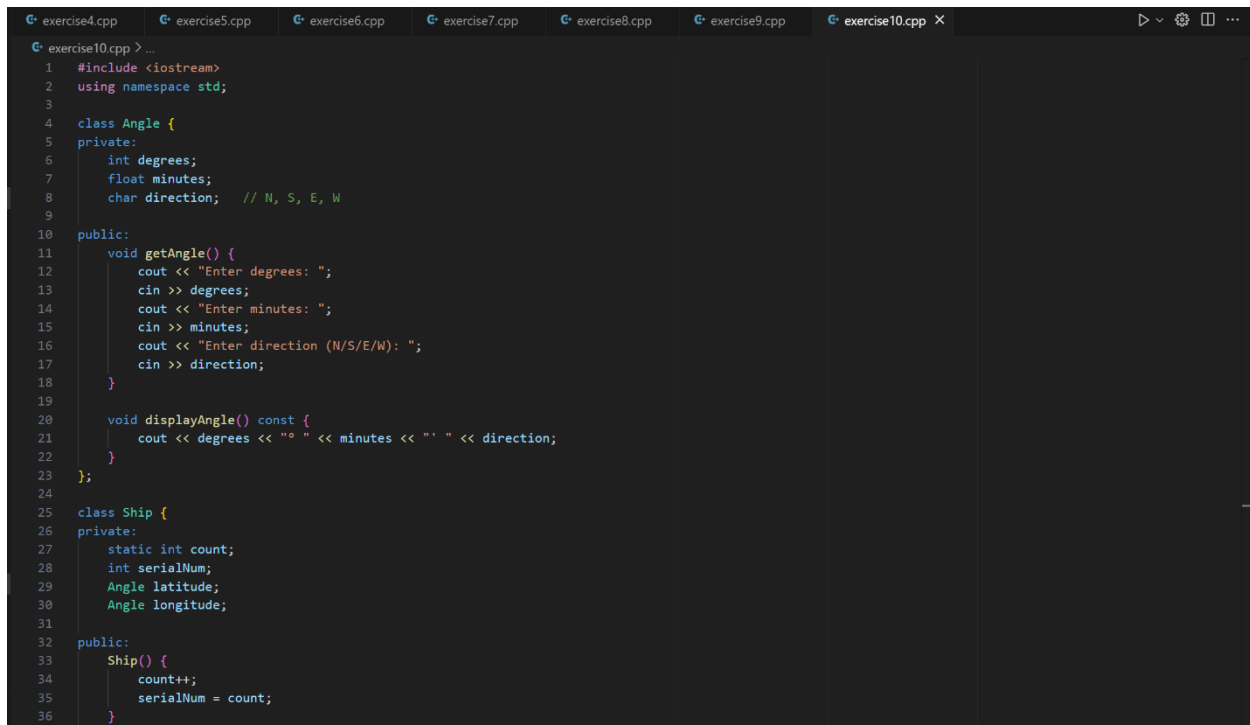
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Active code page: 65001

C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>cd "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\" && g++ exercise9.cpp -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>
```

## Exercise 6.10:

### Code:



```
exercise4.cpp exercise5.cpp exercise6.cpp exercise7.cpp exercise8.cpp exercise9.cpp exercise10.cpp X
exercise10.cpp > ...
1 #include <iostream>
2 using namespace std;
3
4 class Angle {
5 private:
6     int degrees;
7     float minutes;
8     char direction; // N, S, E, W
9
10 public:
11     void getAngle() {
12         cout << "Enter degrees: ";
13         cin >> degrees;
14         cout << "Enter minutes: ";
15         cin >> minutes;
16         cout << "Enter direction (N/S/E/W): ";
17         cin >> direction;
18     }
19
20     void displayAngle() const {
21         cout << degrees << "° " << minutes << "' " << direction;
22     }
23 };
24
25 class Ship {
26 private:
27     static int count;
28     int serialNum;
29     Angle latitude;
30     Angle longitude;
31
32 public:
33     Ship() {
34         count++;
35         serialNum = count;
36     }
37 }
```

```
38     void getPosition() {
39         cout << "Enter Latitude:\n";
40         latitude.getAngle();
41         cout << "Enter Longitude:\n";
42         longitude.getAngle();
43     }
44
45     void report() const {
46         cout << "\nShip Serial Number: " << serialNum << endl;
47         cout << "Latitude: ";
48         latitude.displayAngle();
49         cout << "\nLongitude: ";
50         longitude.displayAngle();
51         cout << endl;
52     }
53 };
54
55 int Ship::count = 0; // initialize static count
56
57 int main() {
58     Ship ships[3];
59
60     for(int i = 0; i < 3; i++) {
61         cout << "\nEnter details for Ship " << i+1 << ":\n";
62         ships[i].getPosition();
63     }
64
65     cout << "\n--- Ship Details ---\n";
66     for(int i = 0; i < 3; i++) {
67         ships[i].report();
68     }
69
70     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++ exercise10.cpp -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 ---
```

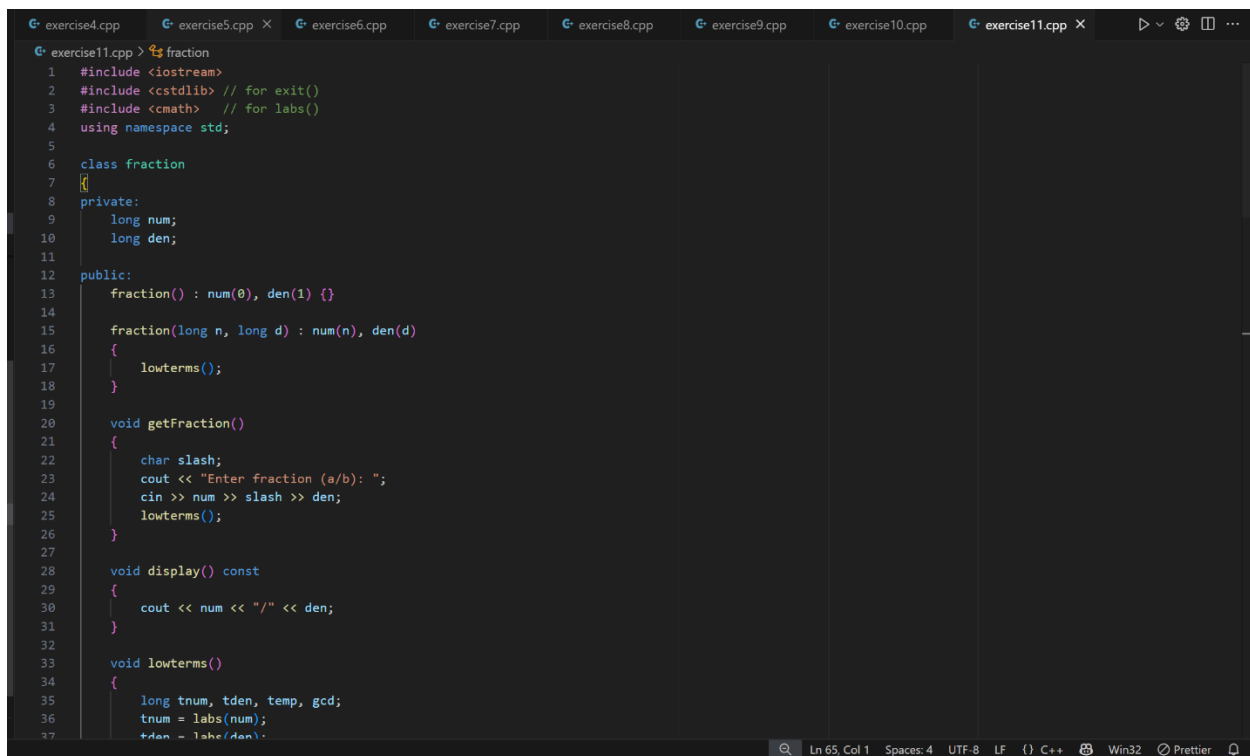
```
Ship Serial Number: 1  
Latitude: 12° 23' S  
Longitude: 98° 89' W
```

```
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:



```
exercise4.cpp exercise5.cpp X exercise6.cpp exercise7.cpp exercise8.cpp exercise9.cpp exercise10.cpp exercise11.cpp X
1 #include <iostream>
2 #include <cstdlib> // for exit()
3 #include <cmath> // for labs()
4 using namespace std;
5
6 class fraction
7 {
8 private:
9     long num;
10    long den;
11
12 public:
13    fraction() : num(0), den(1) {}
14
15    fraction(long n, long d) : num(n), den(d)
16    {
17        lowterms();
18    }
19
20    void getFraction()
21    {
22        char slash;
23        cout << "Enter fraction (a/b): ";
24        cin >> num >> slash >> den;
25        lowterms();
26    }
27
28    void display() const
29    {
30        cout << num << "/" << den;
31    }
32
33    void lowterms()
34    {
35        long tnum, tden, temp, gcd;
36        tnum = labs(num);
37        tden = labs(den);
```

```

37     tden = labs(den);
38     if (tden == 0)
39     {
40         cout << "Illegal fraction: division by 0\n";
41         exit(1);
42     }
43     else if (tnum == 0)
44     { // check for 0/n
45         num = 0;
46         den = 1;
47         return;
48     }
49
50     while (tnum != 0)
51     {
52         if (tnum < tden)
53         {
54             temp = tnum;
55             tnum = tden;
56             tden = temp;
57         }
58         tnum = tnum - tden;
59     }
60     gcd = tden;
61
62     num = num / gcd;
63     den = den / gcd;
64 }
65
66 fraction operator+(const fraction &f) const
67 {
68     return fraction(num * f.den + den * f.num, den * f.den);
69 }
70

```

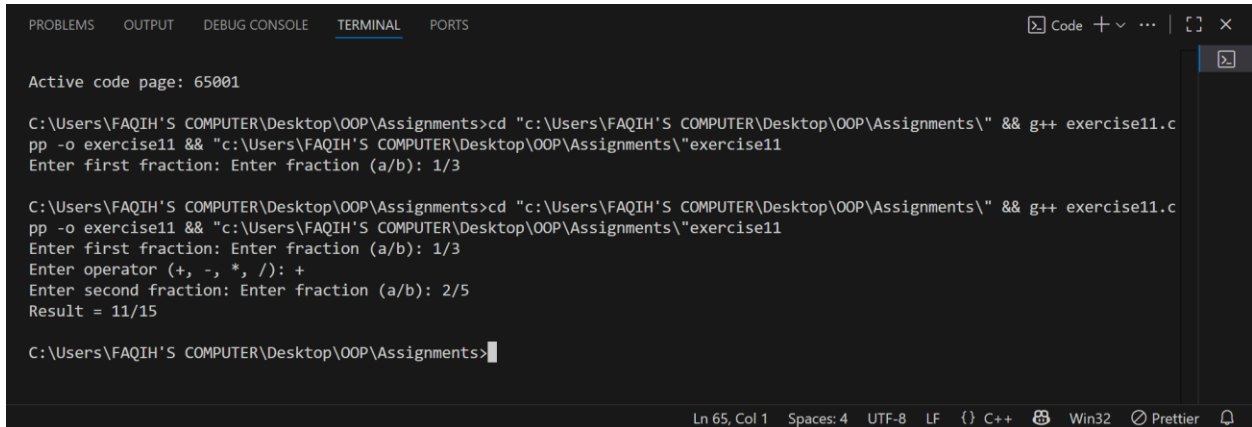
```

70
71 fraction operator-(const fraction &f) const
72 {
73     return fraction(num * f.den - den * f.num, den * f.den);
74 }
75
76 fraction operator*(const fraction &f) const
77 {
78     return fraction(num * f.num, den * f.den);
79 }
80
81 fraction operator/(const fraction &f) const
82 {
83     return fraction(num * f.den, den * f.num);
84 }
85 };
86
87 int main()
88 {
89     fraction f1, f2, result;
90     char op;
91
92     cout << "Enter first fraction: ";
93     f1.getFraction();
94     cout << "Enter operator (+, -, *, /): ";
95     cin >> op;
96     cout << "Enter second fraction: ";
97     f2.getFraction();
98
99     switch (op)
100     {
101     case '+':
102         result = f1 + f2;
103         break;

```

```
102     result = f1 + f2;
103     break;
104     case '-':
105     result = f1 - f2;
106     break;
107     case '*':
108     result = f1 * f2;
109     break;
110     case '/':
111     result = f1 / f2;
112     break;
113     default:
114     cout << "Unknown operator\n";
115     exit(1);
116     }
117
118     cout << "Result = ";
119     result.display();
120     cout << endl;
121
122     return 0;
123 }
124
```

## 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++ exercise11.c
pp -o exercise11 && "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\"exercise11
Enter first fraction: Enter fraction (a/b): 1/3

C:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments>cd "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\" && g++ exercise11.c
pp -o exercise11 && "c:\Users\FAQIH'S COMPUTER\Desktop\OOP\Assignments\"exercise11
Enter first fraction: Enter fraction (a/b): 1/3
Enter operator (+, -, *, /): +
Enter second fraction: Enter fraction (a/b): 2/5
Result = 11/15

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

## Exercise 6.12:

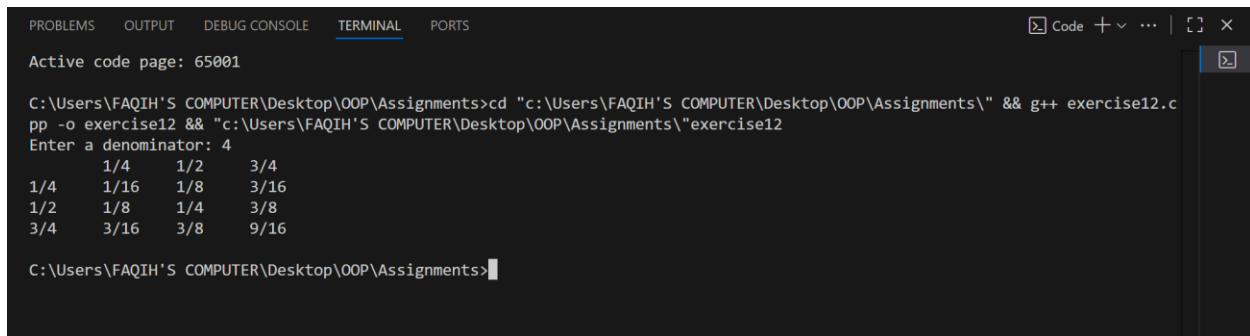
### Code:

```

exercise12.cpp > main()
1 #include <iostream>
2 #include <cstdlib>
3 #include <cmath>
4 using namespace std;
5
6 class fraction {
7 private:
8     long num, den;
9
10 public:
11     fraction() : num(0), den(1) {}
12     fraction(long n, long d) : num(n), den(d) {
13         lowterms();
14     }
15
16     void display() const {
17         cout << num << "/" << den;
18     }
19
20     void lowterms() {
21         long tnum, tden, temp, gcd;
22         tnum = labs(num);
23         tden = labs(den);
24         if (tden == 0) {
25             cout << "Illegal fraction: division by 0\n";
26             exit(1);
27         } else if (tnum == 0) {
28             num = 0; den = 1;
29             return;
30         }
31         while (tnum != 0) {
32             if (tnum < tden) {
33                 temp = tnum;
34                 tnum = tden;
35                 tden = temp;
36
37                 tnum = tnum - tden;
38             }
39             gcd = tden;
40             num /= gcd;
41             den /= gcd;
42         }
43
44         fraction operator * (const fraction& f) const {
45             return fraction(num * f.num, den * f.den);
46         };
47     int main() {
48         int denom;
49         cout << "Enter a denominator: ";
50         cin >> denom;
51         cout << "\n";
52         for (int i = 1; i < denom; i++) {
53             fraction f(i, denom);
54             f.display();
55             cout << "\t";
56         }
57         cout << endl;
58         for (int i = 1; i < denom; i++) {
59             fraction rowFrac(i, denom);
60             rowFrac.display();
61             cout << "\t";
62             for (int j = 1; j < denom; j++) {
63                 fraction colFrac(j, denom);
64                 fraction result = rowFrac * colFrac;
65                 result.display();
66                 cout << "\t";
67             }
68             cout << endl;
69         }
70         return 0;
71     }

```

**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++ 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>
```