

Homework. Part I – Review Questions.

Chapter 5. Page 311 – 312. Qn. 1 – 4, 14 – 16, 19 – 26.

1. To **increment** a value means to increase it by one.
2. To **decrement** a value means to decrease it by one.
3. When the increment or decrement operator is placed before the operand (or to the operand's left), the operator is being used in **prefix** mode.
4. When the increment or decrement operator is placed after the operand (or to the operand's right), the operator is being used in **postfix** mode.
14. The **for** loop is ideal for situations that require a counter.
15. The **do-while** loop always iterates at least once.
16. The **while** and **for** loops will not iterate at all if their test expressions are false to start with.
19. The **break** statement causes a loop to terminate immediately.
20. The **continue** statement causes a loop to skip the remaining statements in the current iteration.
21. What header file do you need to include in a program that performs file operations? **fstream**
22. What data type do you use when you want to create a file stream object that can write data to a file? **ofstream**
23. What happens if you open an output file and the file already exists? **The existing file will be erased and new file will be created.**
24. What data type do you use when you want to create a file stream object that can read data from a file? **ifstream**

Homework. Part II – Programming Challenge.

Chapter 5. Page 321. Qn. 26 – Using Files – Total and Average Rainfall

Files: main.cpp
 RainfallReader.h
 RainfallReader.cpp
 Rainfall.txt

Screenshot of Runtime:

```
During the months of June-September,  
the total rainfall was 6.94 inches  
and the average monthly rainfall was 1.73 inches
```

```
Process finished with exit code 0
```

Source Files:

main.cpp

```
#include "RainfallReader.h"
```

```
int main() {  
    RainfallReader rr;  
    rr.extractData();  
    rr.printOutput();  
  
    return 0;  
}
```

Header File: RainfallReader.h

```
#include <string>  
using namespace std;  
  
class RainfallReader {  
  
public:  
    string month_start, month_end;  
    double rainfall_total, rainfall_average;  
  
    void extractData();  
    void printOutput();  
  
};
```

Implementation File: RainfallReader.cpp

```
#include <iostream>  
#include <iomanip>  
#include <fstream>  
#include <string>  
#include "RainfallReader.h"  
using namespace std;  
  
void RainfallReader::extractData() {  
    ifstream source;  
    int rain_count = 0;  
    double temp_rain;  
    rainfall_total = 0;  
  
    source.open("Rainfall.txt");  
    source >> month_start;  
    source >> month_end;  
  
    while (source >> temp_rain) {  
        rainfall_total += temp_rain;  
        rain_count++;  
    }  
}
```

```

    rainfall_average = rainfall_total / rain_count;
    source.close();

    return;
}

void RainfallReader::printOutput() {
    cout << "During the months of "
         << month_start << "-"
         << month_end << ", " << endl;

    cout << fixed << showpoint << setprecision(2);
    cout << "the total rainfall was " << rainfall_total << " inches " << endl
         << "and the average monthly rainfall was "
         << rainfall_average << " inches " << endl;

    return;
}

```

Data File: Rainfall.txt

June

September

2.35	1.15	2.03	1.41
------	------	------	------