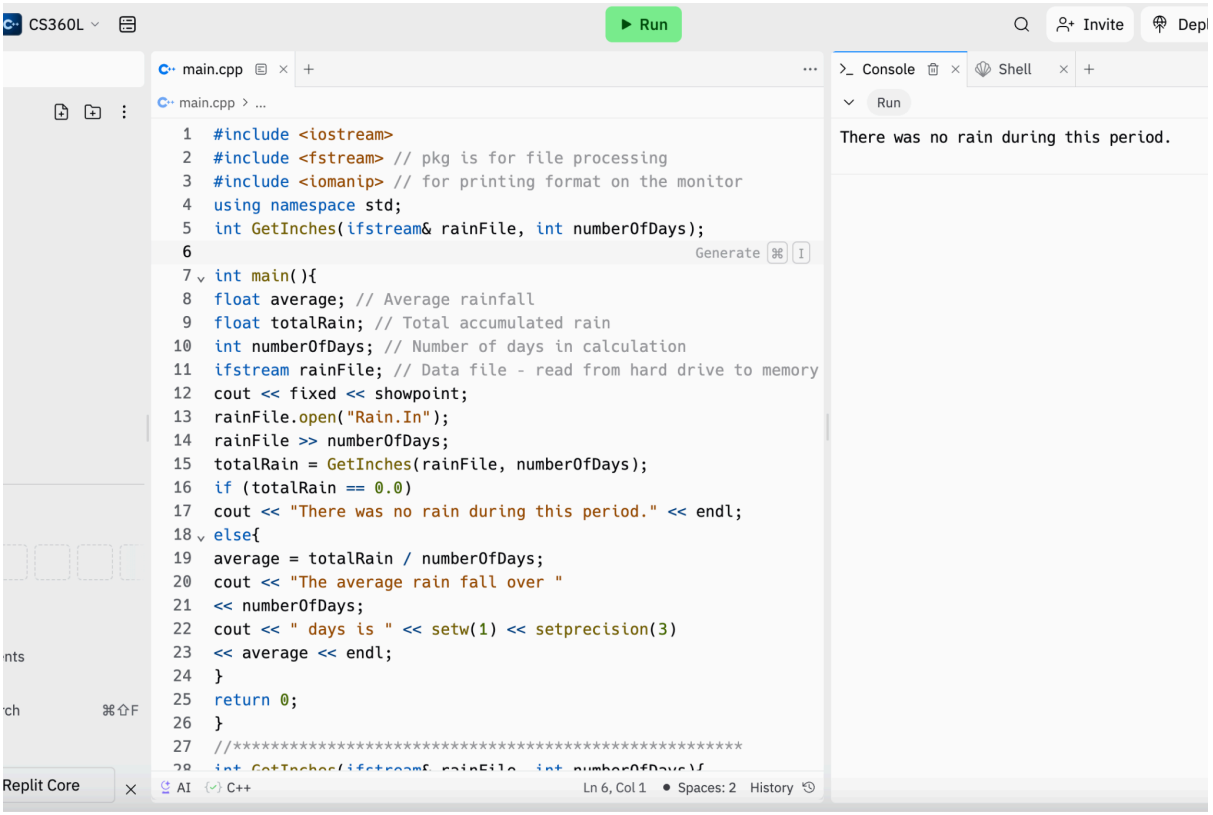
Ronish Shrestha(19707)
CS360Lab

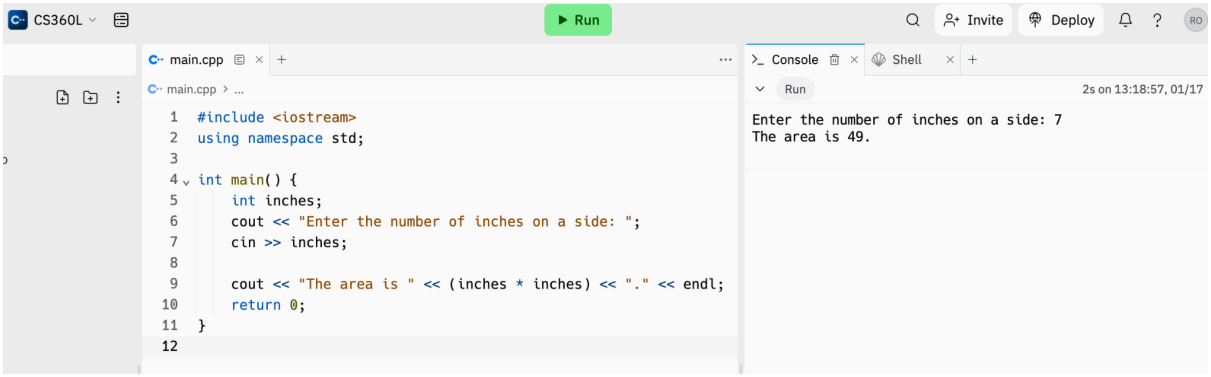
1.

```
#include <iostream>
#include <fstream> // pkg is for file processing
#include <iomanip> // for printing format on the monitor
using namespace std;
int GetInches(ifstream& rainFile, int numberOfDays);

int main(){
float average; // Average rainfall
float totalRain; // Total accumulated rain
int numberOfDays; // Number of days in calculation
ifstream rainFile; // Data file – read from hard drive to memory
cout << fixed << showpoint;
rainFile.open("Rain.In");
rainFile >> numberOfDays;
totalRain = GetInches(rainFile, numberOfDays);
if (totalRain == 0.0)
cout << "There was no rain during this period." << endl;
else{
average = totalRain / numberOfDays;
cout << "The average rain fall over "
<< numberOfDays;
cout << " days is " << setw(1) << setprecision(3)
<< average << endl;
}
return 0;
}
//*******************************************************
int GetInches(ifstream& rainFile, int numberOfDays){
float inches; // Day's worth of rain
int counter; // Loop control variable
float totalRain = 0.0;
counter = 1;
while (counter <= numberOfDays){
rainFile >> inches;
totalRain = totalRain + inches;
counter++;
}
return totalRain;
}
```



2.

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

int main() {
    int inches;
    cout << "Enter the number of inches on a side: ";
    cin >> inches;

    cout << "The area is " << (inches * inches) << "." << endl;
    return 0;
}
```

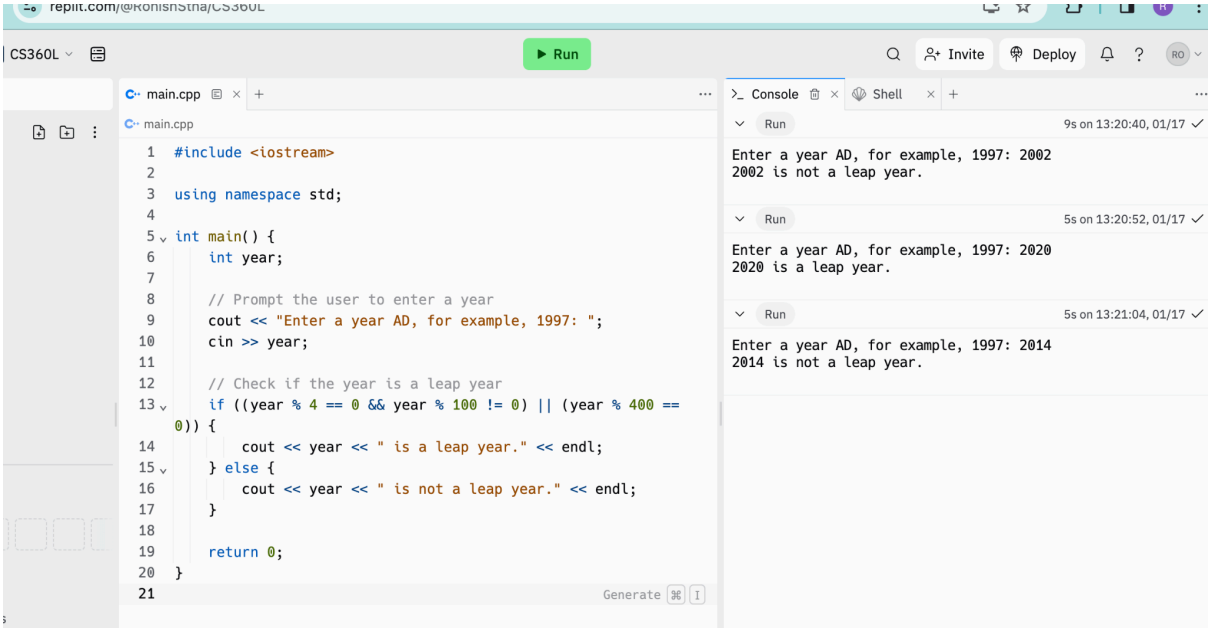


int inches; declares an integer variable named inches.
cout << "Enter the number of inches on a side " << endl; outputs a message prompting the user to enter the side length.
cin >> inches; takes the user's input and stores it in the inches variable.
cout << endl << "The area is " << inches * inches << "." << endl; calculates the area by squaring the value of inches and outputs the result.
return 0; indicates that the program

3.

```
#include <iostream>

using namespace std;

int main() {
    int year;

    // Prompt the user to enter a year
    cout << "Enter a year AD, for example, 1997: ";
    cin >> year;

    // Check if the year is a leap year
    if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
        cout << year << " is a leap year." << endl;
    } else {
```

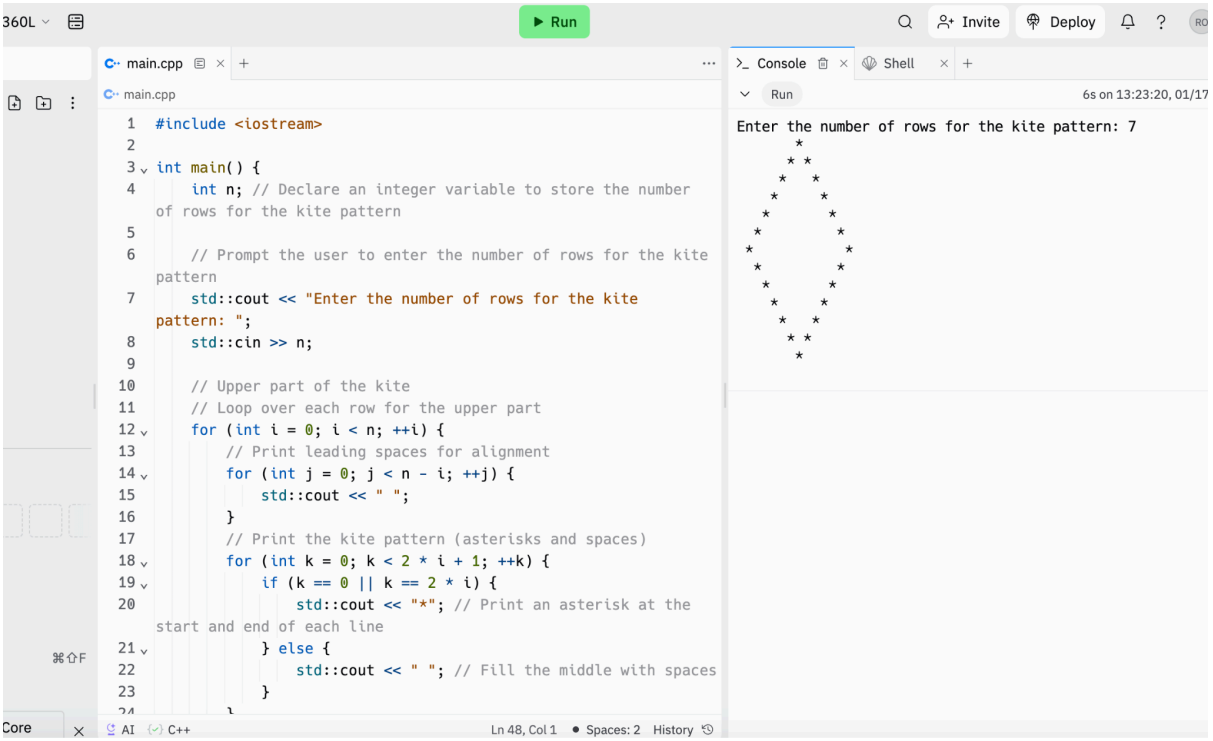
```
        cout << year << " is not a leap year." << endl;
    }

    return 0;
}
```



4.

```
#include <iostream>

int main() {
    int n; // Declare an integer variable to store the number of rows for the kite pattern

    // Prompt the user to enter the number of rows for the kite pattern
    std::cout << "Enter the number of rows for the kite pattern: ";
    std::cin >> n;

    // Upper part of the kite
    // Loop over each row for the upper part
    for (int i = 0; i < n; ++i) {
        // Print leading spaces for alignment
        for (int j = 0; j < n - i; ++j) {
            std::cout << " ";
        }
        // Print the kite pattern (asterisks and spaces)
        for (int k = 0; k < 2 * i + 1; ++k) {
            if (k == 0 || k == 2 * i) {
```

```
            std::cout << "*"; // Print an asterisk at the start and end of each line
        } else {
            std::cout << " "; // Fill the middle with spaces
        }
    }
    std::cout << std::endl; // Move to the next line
}

// Lower part of the kite
// Loop over each row for the lower part (in reverse order)
for (int i = n - 2; i >= 0; --i) {
    // Print leading spaces for alignment
    for (int j = 0; j < n - i; ++j) {
        std::cout << " ";
    }
    // Print the kite pattern (asterisks and spaces)
    for (int k = 0; k < 2 * i + 1; ++k) {
        if (k == 0 || k == 2 * i) {
            std::cout << "*"; // Print an asterisk at the start and end of each line
        } else {
            std::cout << " "; // Fill the middle with spaces
        }
    }
    std::cout << std::endl; // Move to the next line
}

return 0; // Indicate successful program termination
}
```


360L
Run
Invite
Deploy
main.cpp
main.cpp
Console
Shell
Run
6s on 13:23:20, 01/17
Enter the number of rows for the kite pattern: 7
#include <iostream>
int main() {
int n; // Declare an integer variable to store the number of rows for the kite pattern
// Prompt the user to enter the number of rows for the kite pattern
std::cout << "Enter the number of rows for the kite pattern: ";
std::cin >> n;
// Upper part of the kite
// Loop over each row for the upper part
for (int i = 0; i < n; ++i) {
// Print leading spaces for alignment
for (int j = 0; j < n - i; ++j) {
std::cout << " ";
}
// Print the kite pattern (asterisks and spaces)
for (int k = 0; k < 2 * i + 1; ++k) {
if (k == 0 || k == 2 * i) {
std::cout << "*"; // Print an asterisk at the start and end of each line
} else {
std::cout << " "; // Fill the middle with spaces
}
AI
C++
Ln 48, Col 1
Spaces: 2
History
Core