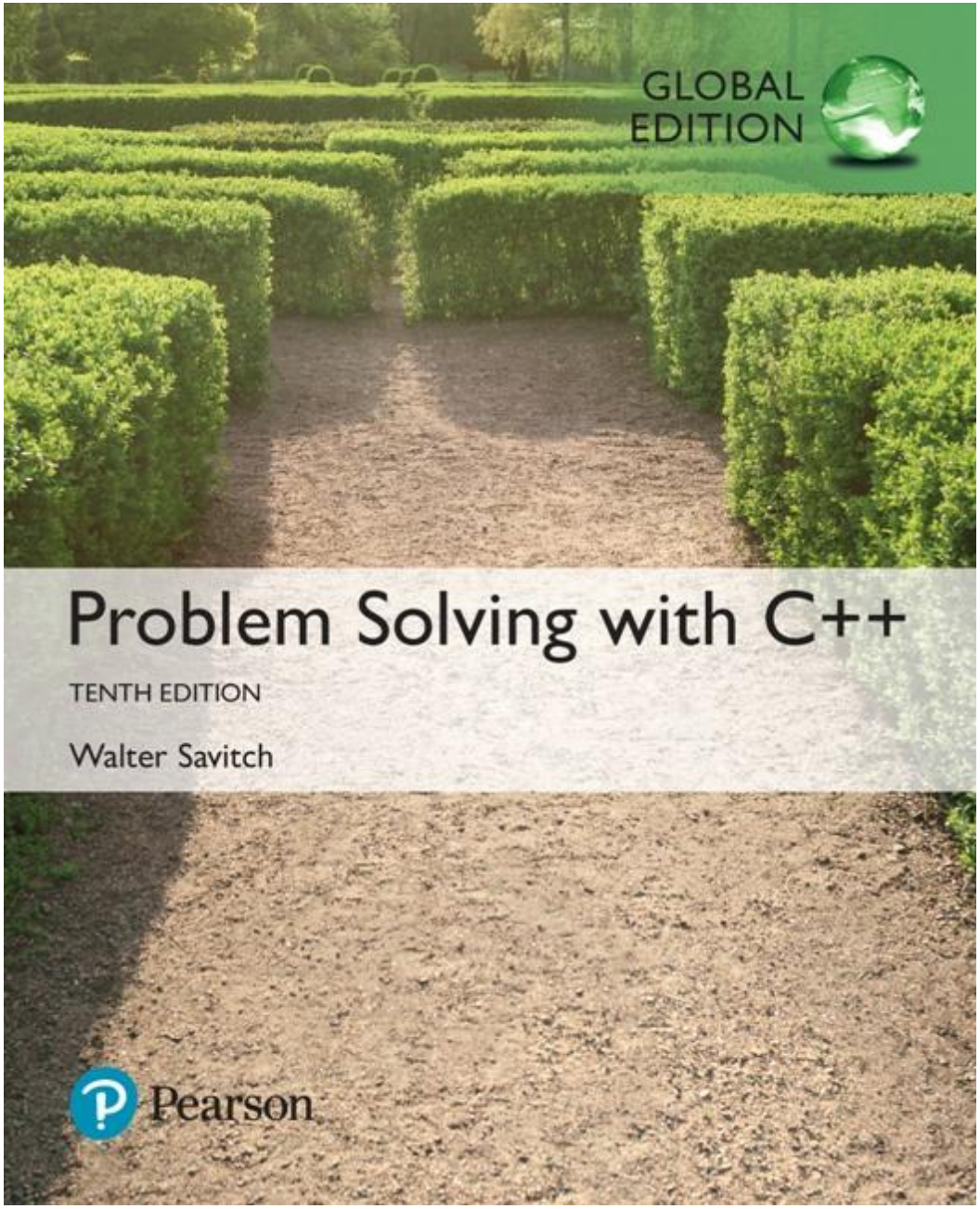
Ahram Canadian University

Fundamental of programming 1

Dr. Marian Wagdy

Lecture 3


GLOBAL
EDITION
Problem Solving with C++
TENTH EDITION
Walter Savitch
Pearson

Chapter 2

2.1 Variables and Assignments

2.2 Input and Output

2.3 Data Types and Expressions

2.4 Simple Flow of Control

2.5 Program Style

2.3

Data Types and Expressions

Data Types and Expressions

- 2 and 2.0 are not the same number
  - A whole number such as 2 is of type int
  - A real number such as 2.0 is of type double

- Numbers of type int are stored as exact values
- Numbers of type double may be stored as approximate values due to limitations on number of significant digits that can be represented

Writing Integer constants

- Type int does not contain decimal points
  - Examples: 34 45 1 89

Writing Double Constants

- Type double can be written in two ways
  - Simple form must include a decimal point
    - Examples: 34.1 23.0034 1.0 89.9
  - Floating Point Notation (Scientific Notation)
    - Examples: 3.41e1 means 34.1
      3.67e17 means 36700000000000000.0
      5.89e-6 means 0.00000589
  - Number left of e does not require a decimal point
  - Exponent cannot contain a decimal point

Other Number Types

- Various number types have different memory requirements
  - More precision requires more bytes of memory
  - Very large numbers require more bytes of memory
  - Very small numbers require more bytes of memory

Integer types

4 bytes (32 bits)

- long or long int (often 4 bytes)
  - Equivalent forms to declare very large integers

```
long  bigTotal;
long int bigTotal;
```

- short or short int (often 2 bytes)
  - Equivalent forms to declare smaller integers

```
short smallTotal;
short int smallTtotal;
```

Floating point types

8 bytes (64 bits)

- long double (often 10 bytes)
  - Declares floating point numbers with up to 19 significant digits

```
long double bigNumber;
```

- float (often 4 bytes)
  - Declares floating point numbers with up to 7 significant digits

```
float notSoBigNumber;
```

Type char

- Computers process character data too
- char
  - Short for character
  - Can be any single character from the keyboard

- To declare a variable of type char:

```
char letter;
```

char constants

- Character constants are enclosed in single quotes

```
char letter = 'a';
```

- Strings of characters, even if only one character is enclosed in double quotes
  - "a" is a string of characters containing one character
  - 'a' is a value of type character

C++11 Types

- The size of numeric data types can vary from one machine to another.
  - For example, an int might be 32 bits or 64 bits

- C++11 introduced new integer types that specify exactly the size and whether or not the data type is signed or unsigned

- C++11 also has an “auto” type that deduces the type of the variable based on the expression on the right hand side of the assignment

```
auto x = 2.4 * 10;    // x becomes a double due to 2.4
```

Reading Character Data

- cin skips blanks and line breaks looking for data
- The following reads two characters but skips any space that might be between

```
char symbol1, symbol2;
cin  >>  symbol1  >>  symbol2;
```

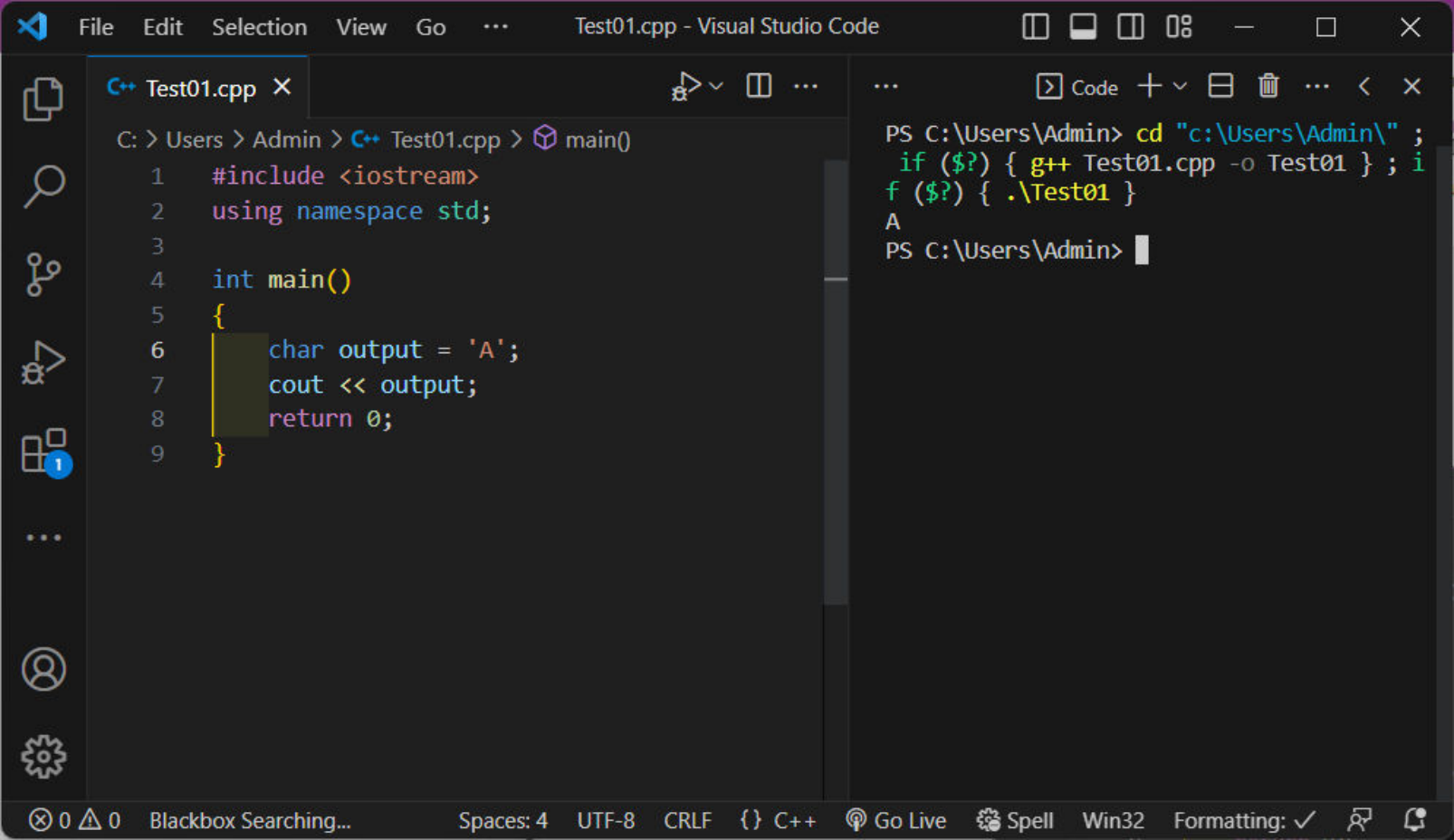
- User normally separate data items by spaces

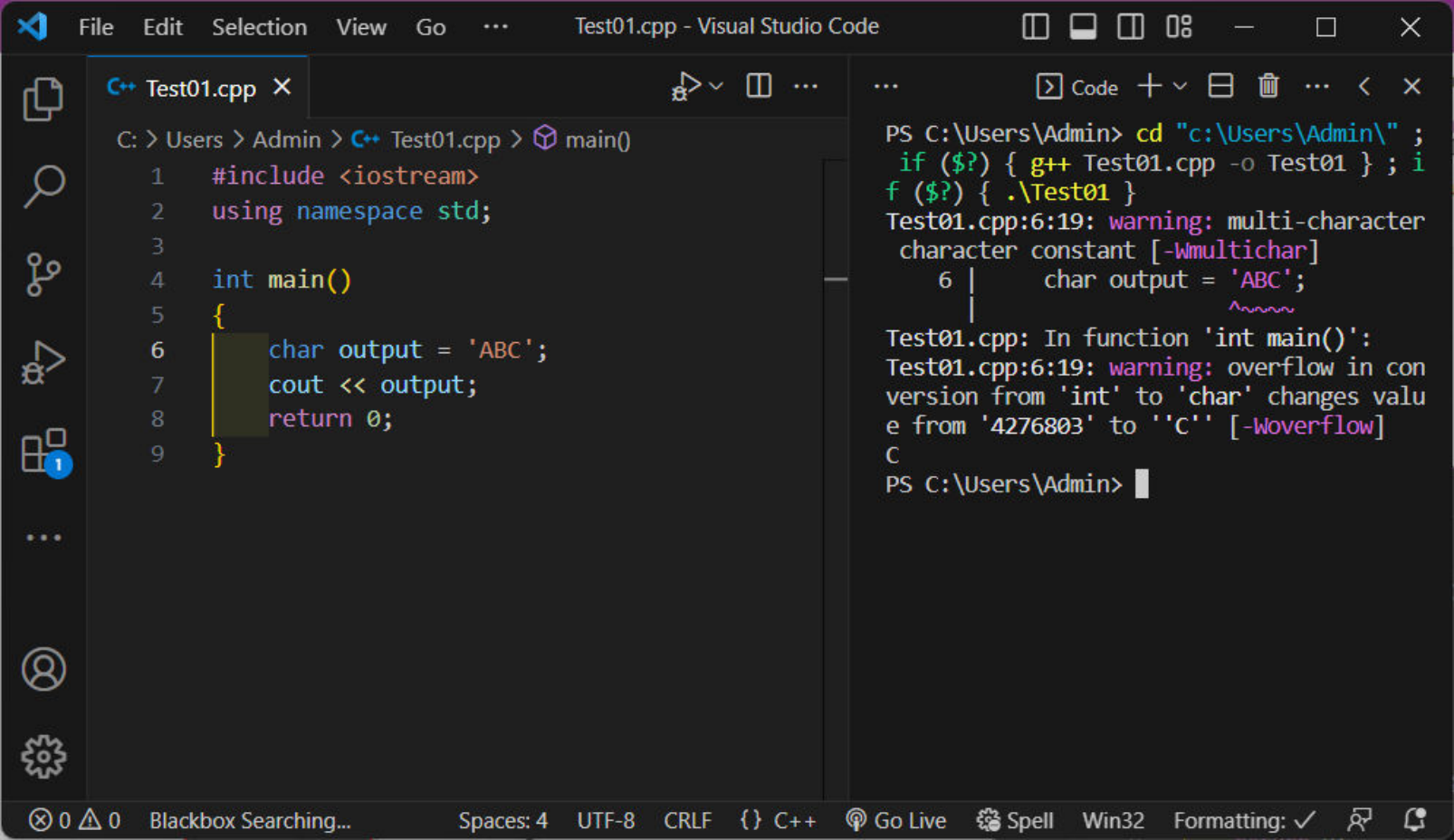
```
J    D
```

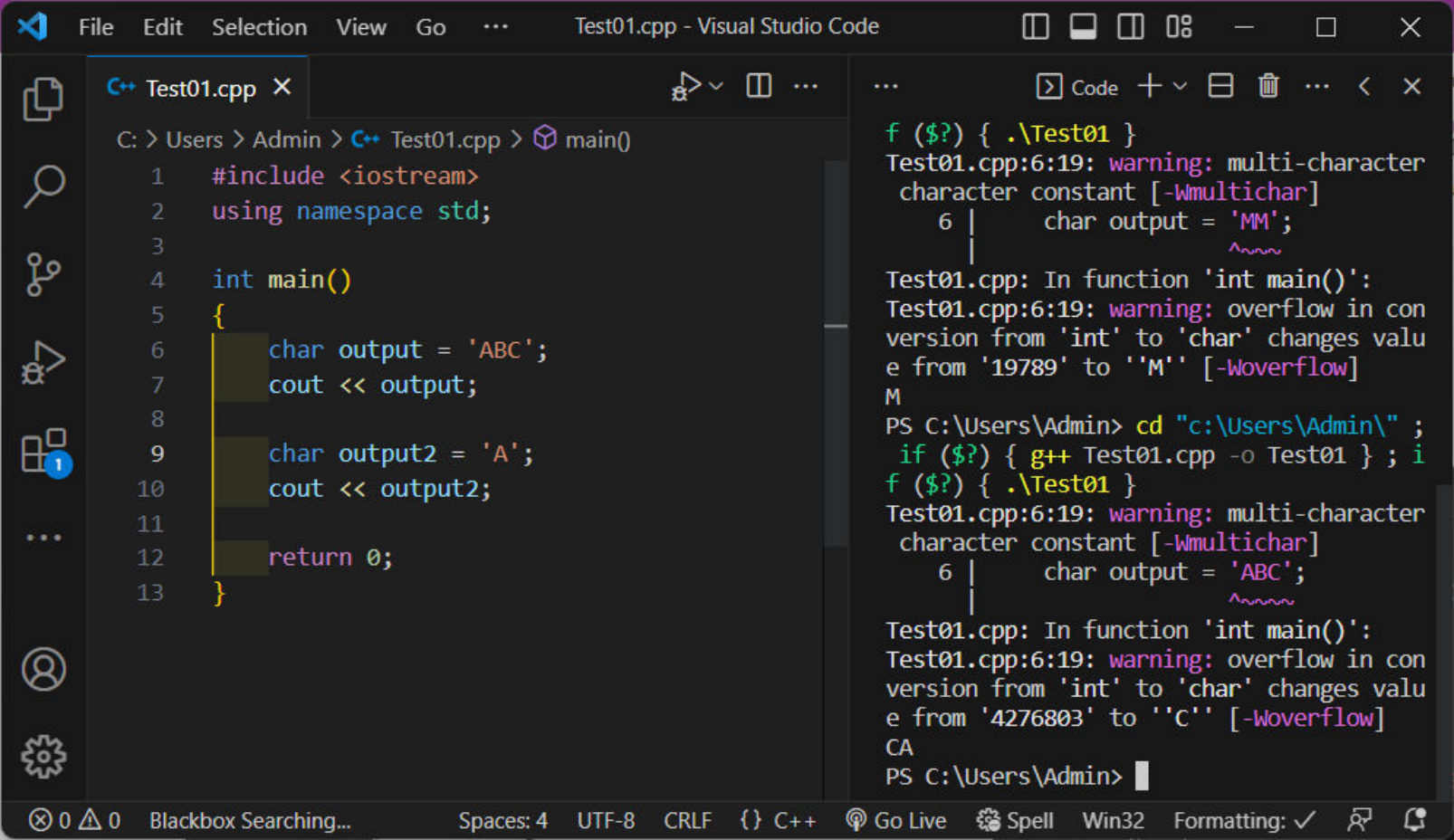
- Results are the same if the data is not separated by spaces

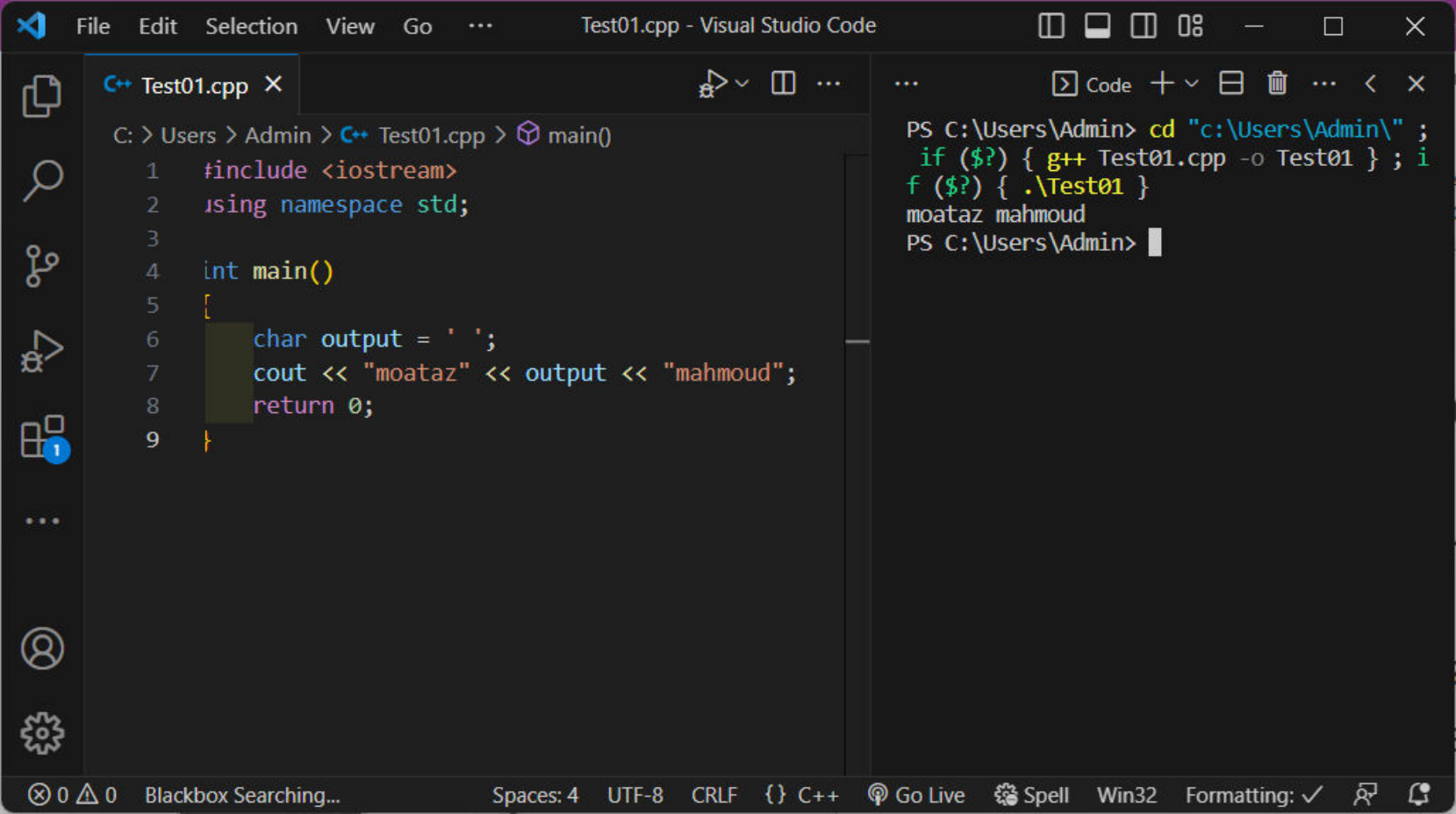
```
JD
```

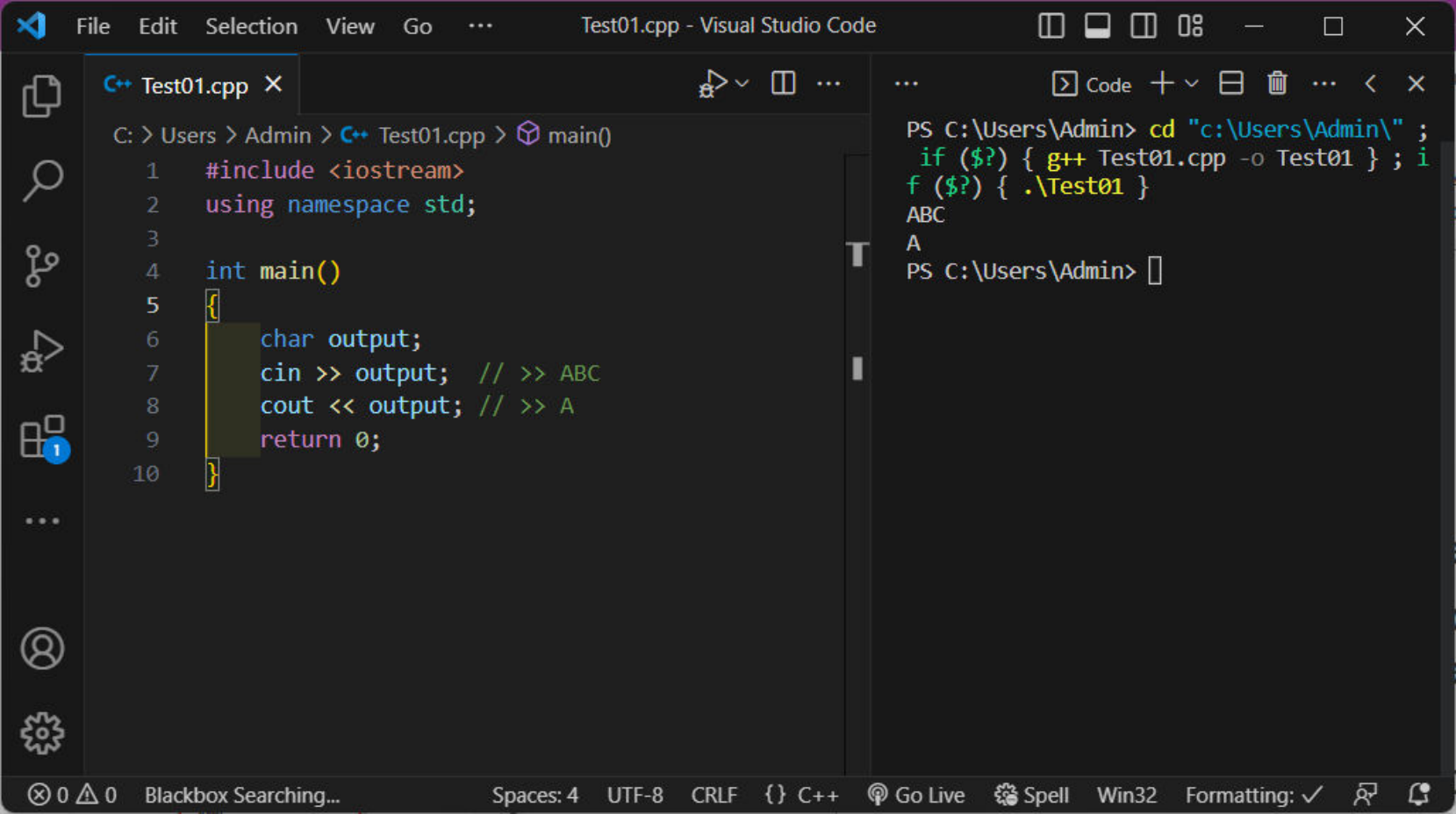
Display 2.4

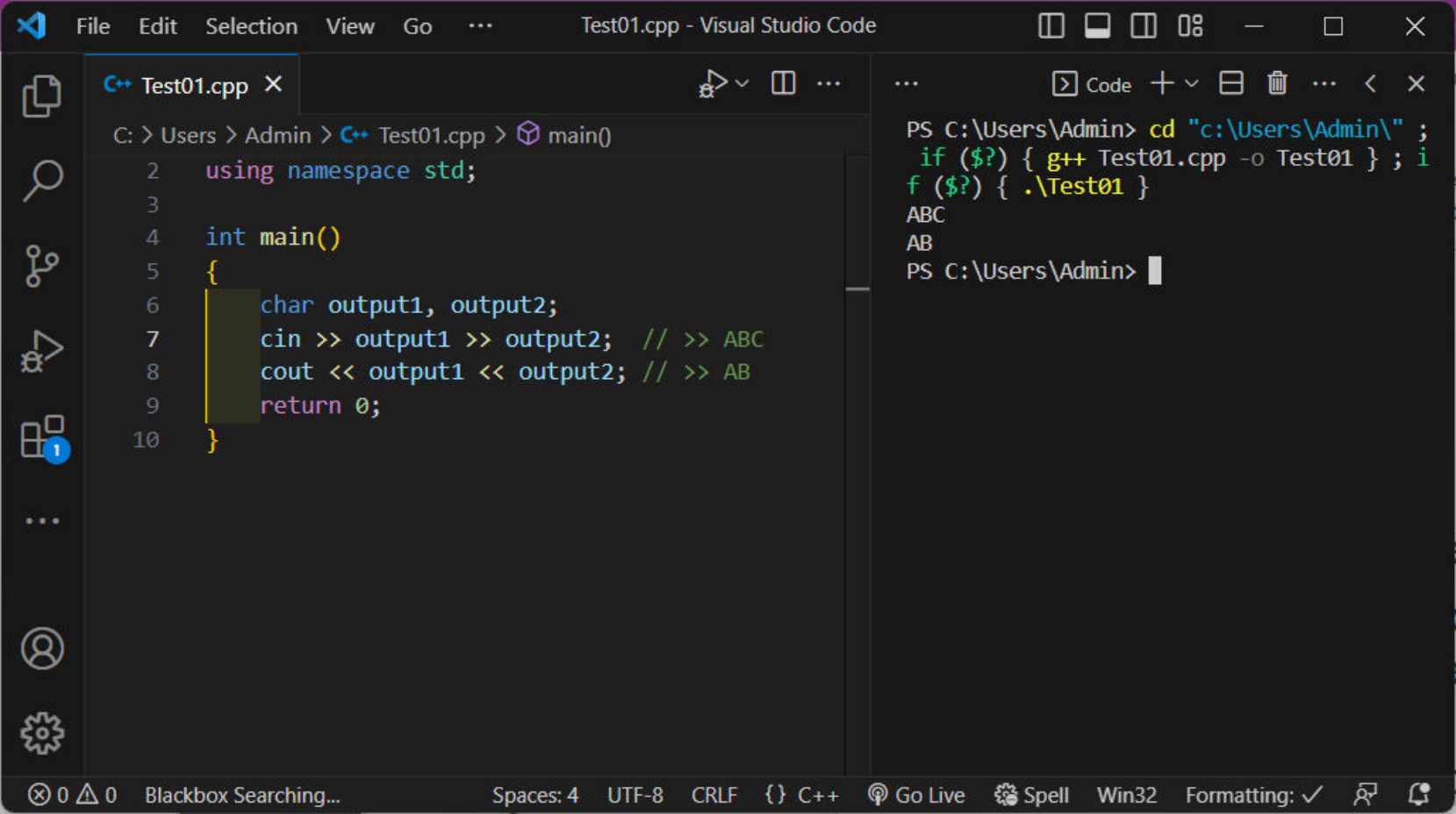

File Edit Selection View Go ··· Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6 char output = 'A';
7 cout << output;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
A
PS C:\Users\Admin>
0 0 Blackbox Searching... Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

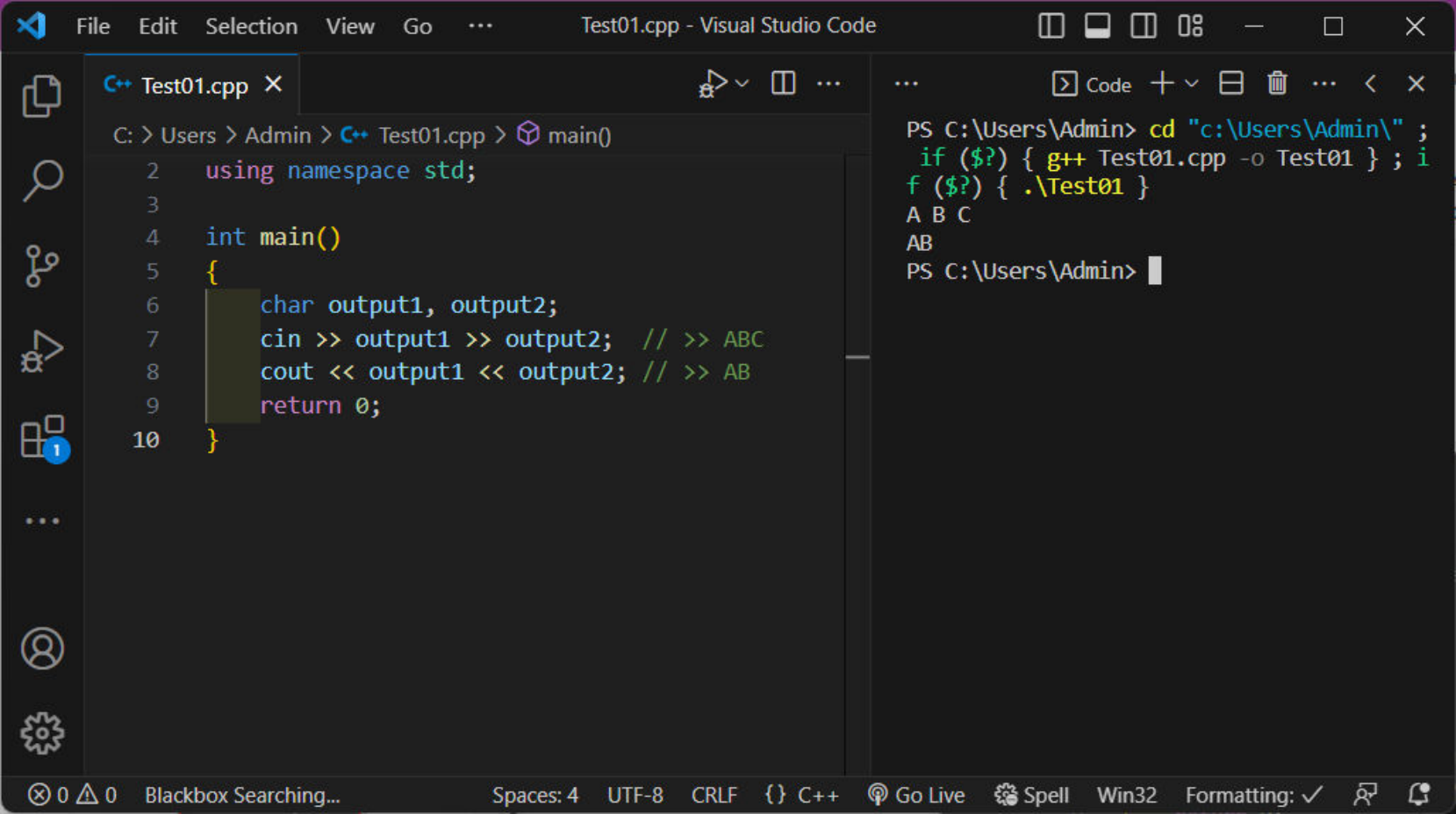

File Edit Selection View Go ··· Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6 char output = 'ABC';
7 cout << output;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Test01.cpp:6:19: warning: multi-character character constant [-Wmultichar]
6 | char output = 'ABC';
| ^~~~~
Test01.cpp: In function 'int main()':
Test01.cpp:6:19: warning: overflow in conversion from 'int' to 'char' changes value from '4276803' to ''C'' [-Woverflow]
C
PS C:\Users\Admin>
0 0 Blackbox Searching... Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6 char output = 'ABC';
7 cout << output;
8
9 char output2 = 'A';
10 cout << output2;
11
12 return 0;
13 }
Code
f ($?) { .\Test01 }
Test01.cpp:6:19: warning: multi-character character constant [-Wmultichar]
6 | char output = 'MM';
| ^~~~
Test01.cpp: In function 'int main()':
Test01.cpp:6:19: warning: overflow in conversion from 'int' to 'char' changes value from '19789' to ''M'' [-Woverflow]
M
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Test01.cpp:6:19: warning: multi-character character constant [-Wmultichar]
6 | char output = 'ABC';
| ^~~~~
Test01.cpp: In function 'int main()':
Test01.cpp:6:19: warning: overflow in conversion from 'int' to 'char' changes value from '4276803' to ''C'' [-Woverflow]
CA
PS C:\Users\Admin>
0 0 Blackbox Searching... Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
#include <iostream>
using namespace std;
int main()
{
char output = ' ';
cout << "moataz" << output << "mahmoud";
return 0;
}
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
moataz mahmoud
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
#include <iostream>
using namespace std;
int main()
{
char output;
cin >> output; // >> ABC
cout << output; // >> A
return 0;
}
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
ABC
A
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
using namespace std;
int main()
{
char output1, output2;
cin >> output1 >> output2; // >> ABC
cout << output1 << output2; // >> AB
return 0;
}
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
ABC
AB
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
using namespace std;
int main()
{
char output1, output2;
cin >> output1 >> output2; // >> ABC
cout << output1 << output2; // >> AB
return 0;
}
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
A B C
AB
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

Type string

- string is a class, different from the primitive data types discussed so far
  - Difference is discussed in Chapter 8
  - Use double quotes around the text to store into the string variable
  - Requires the following be added to the top of your program:

```
#include <string>
```

- To declare a variable of type string:

```
string name = "Apu Nahasapeemapetilon";
```

Display 2.5

Type Compatibilities

- In general store values in variables of the same type
  - This is a type mismatch:

```
int intVariable;
intVariable = 2.99;
```

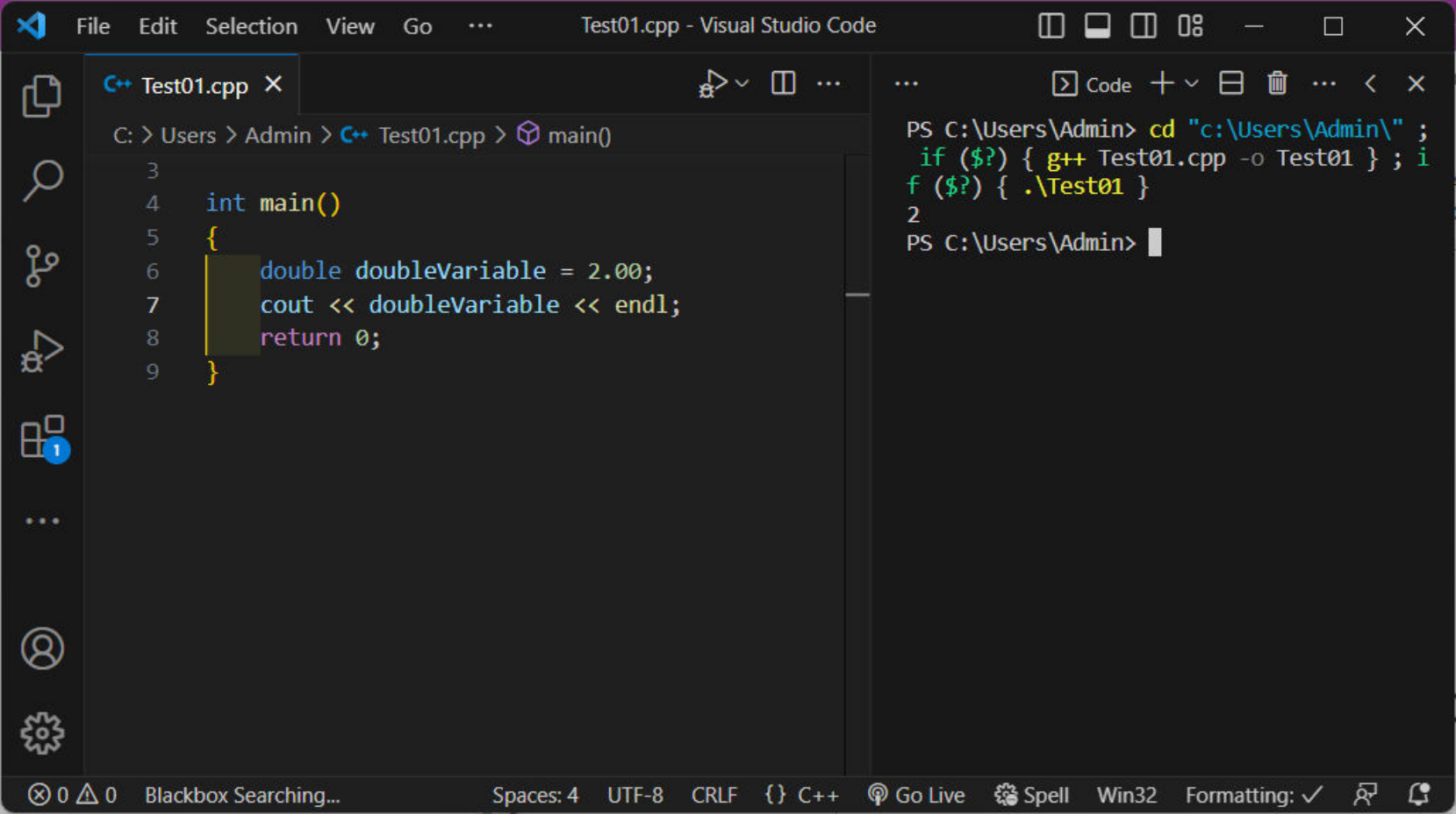
  - If your compiler allows this, intVariable will most likely contain the value 2, not 2.99

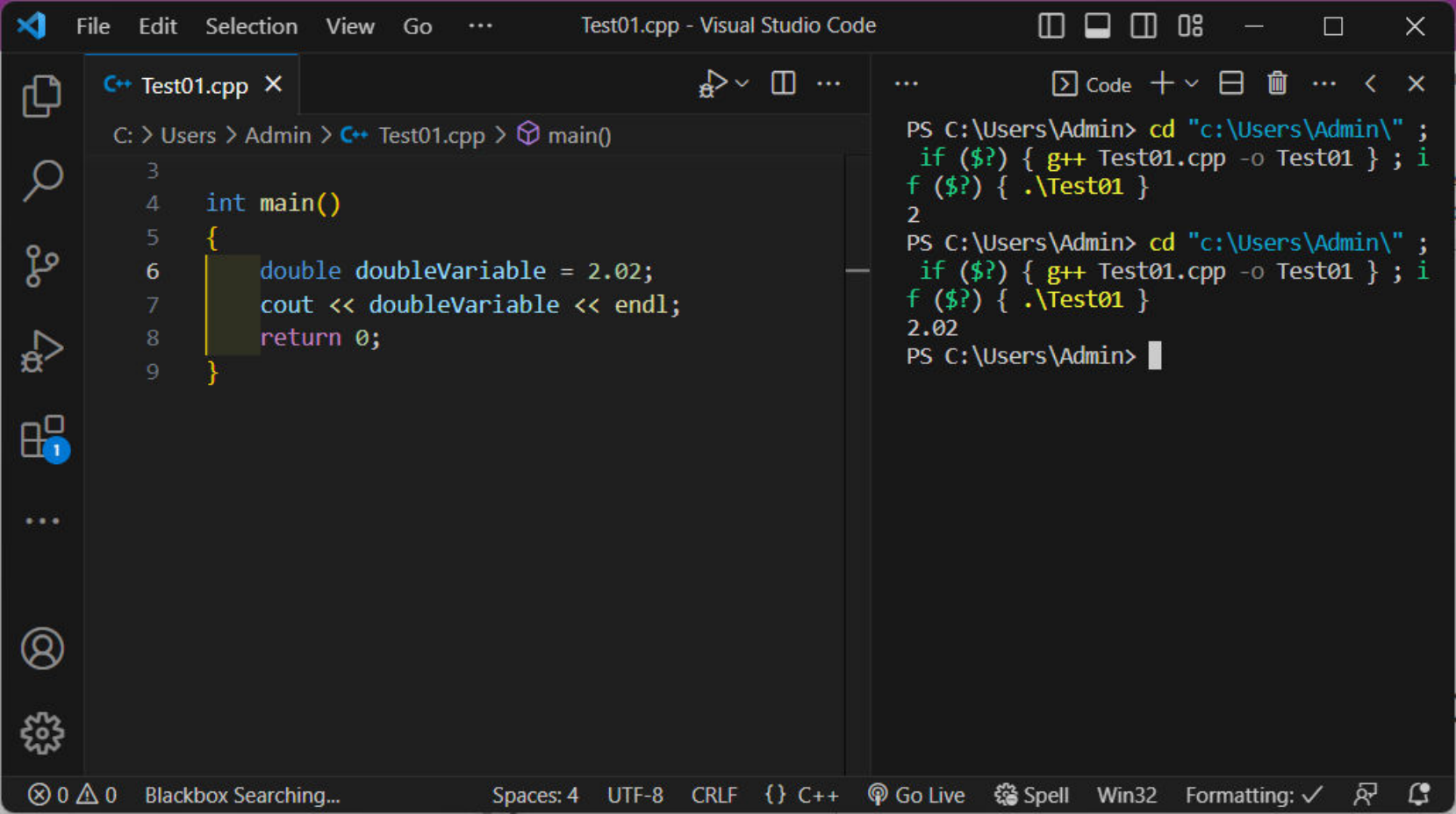
int ←→ double (part 1)

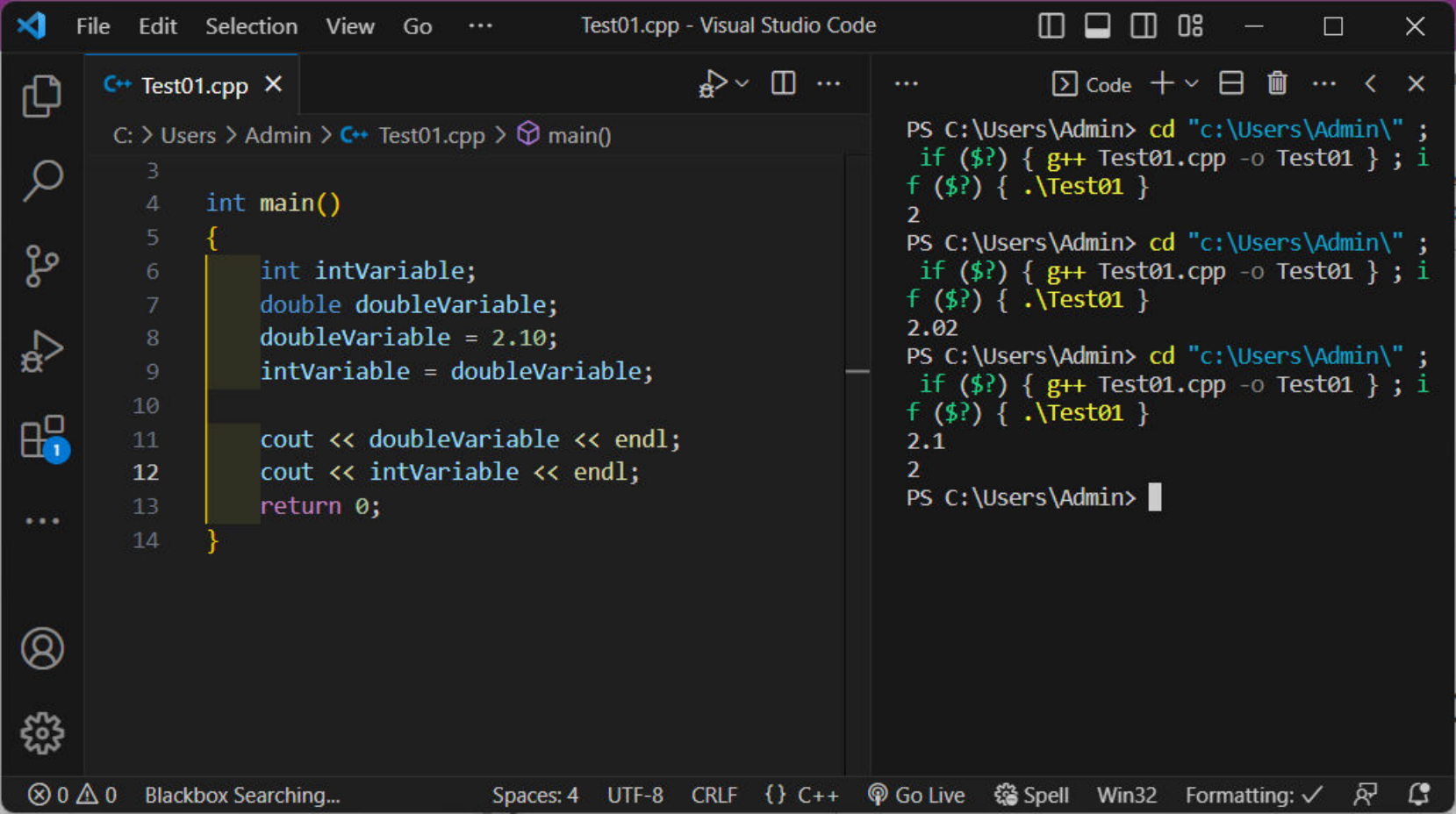
- Variables of type double should not be assigned
  to variables of type int

```
int intVariable;
double doubleVariable;
doubleVariable = 2.00;
 intVariable = doubleVariable;
```

  - If allowed, intVariable contains 2, not 2.00


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 double doubleVariable = 2.00;
7 cout << doubleVariable << endl;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
2
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 double doubleVariable = 2.02;
7 cout << doubleVariable << endl;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
2
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
2.02
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

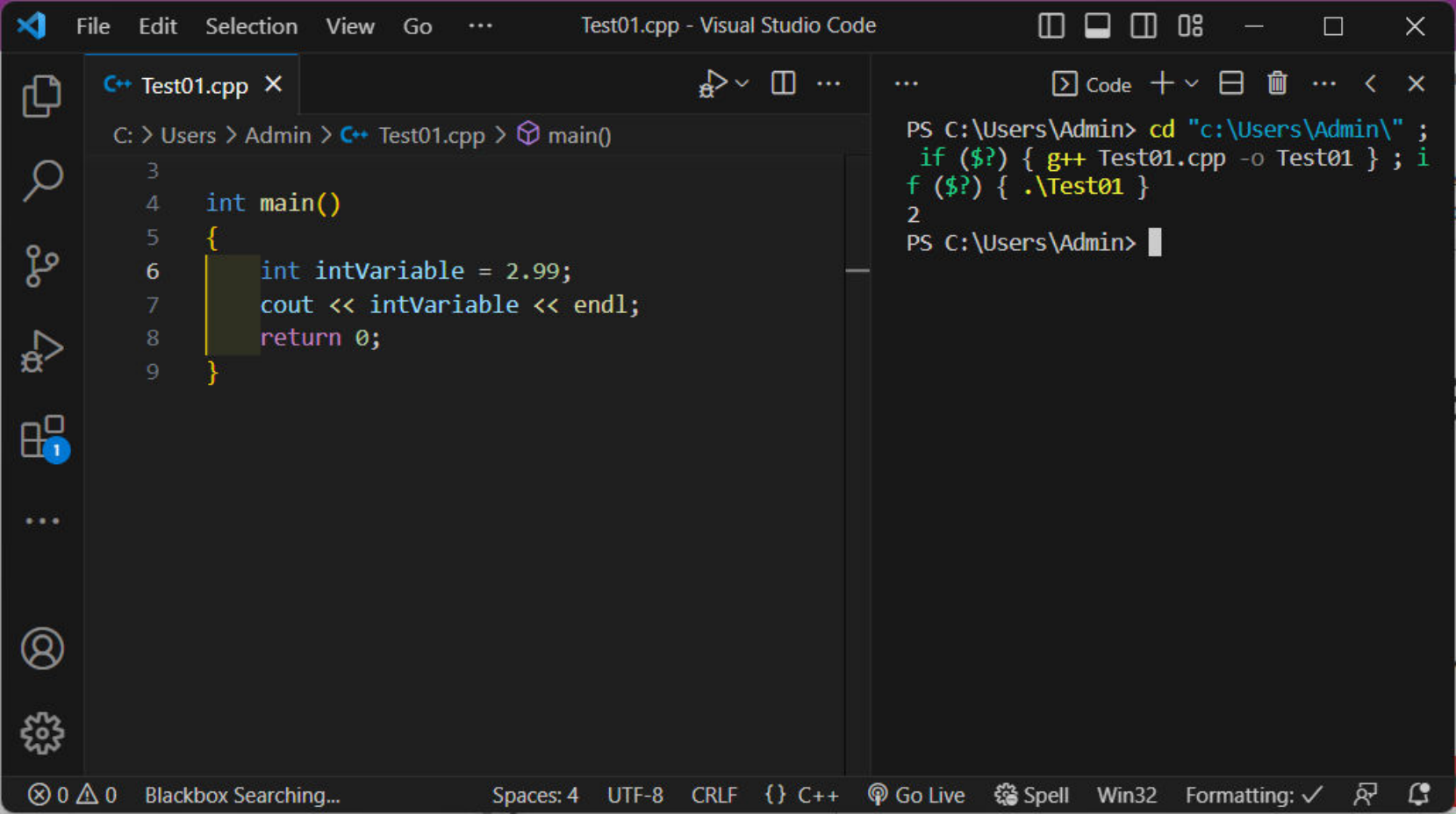

File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 int intVariable;
7 double doubleVariable;
8 doubleVariable = 2.10;
9 intVariable = doubleVariable;
10
11 cout << doubleVariable << endl;
12 cout << intVariable << endl;
13 return 0;
14 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
2
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
2.02
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
2.1
2
PS C:\Users\Admin>
0 0 Blackbox Searching... Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

int ←→ double (part 2)

- Integer values can normally be stored in variables of type double

```
double doubleVariable;
doubleVariable = 2;
```

  - doubleVariable will contain 2.0


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 int intVariable = 2.99;
7 cout << intVariable << endl;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
2
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

char ← → int

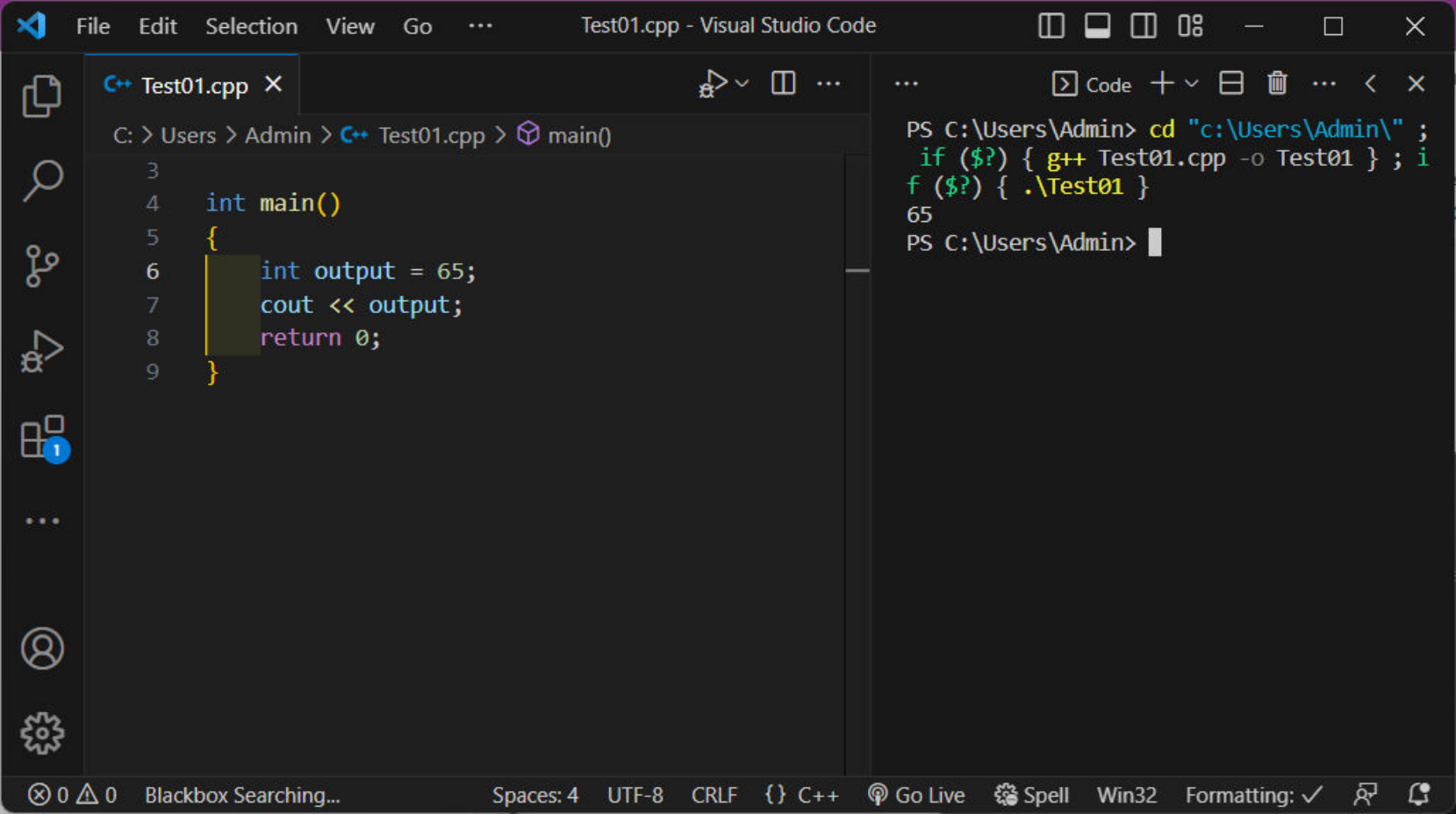
- The following actions are possible but generally not recommended!
- It is possible to store char values in integer variables

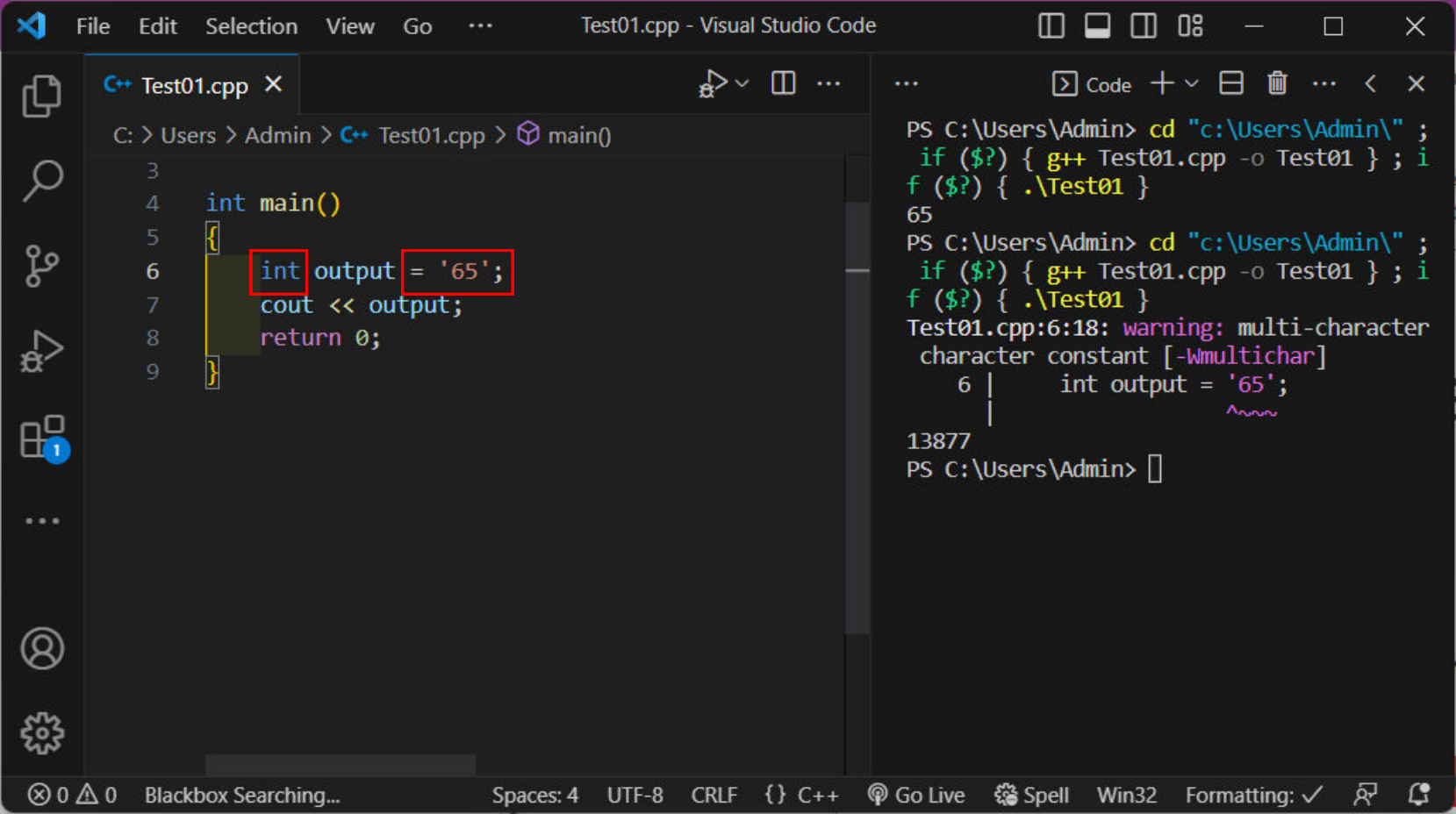
```
int value = 'A';
```

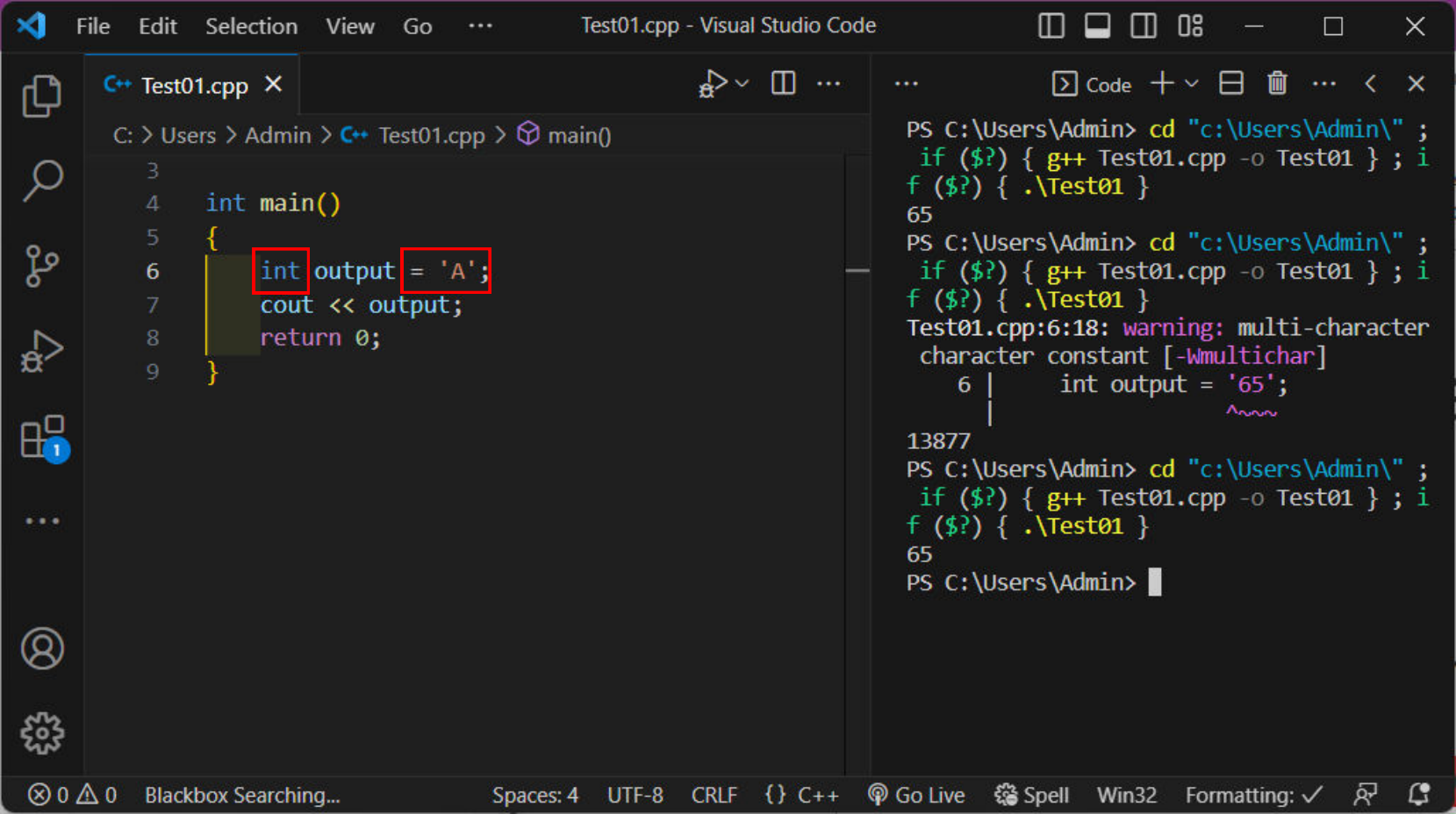
value will contain an integer representing 'A'

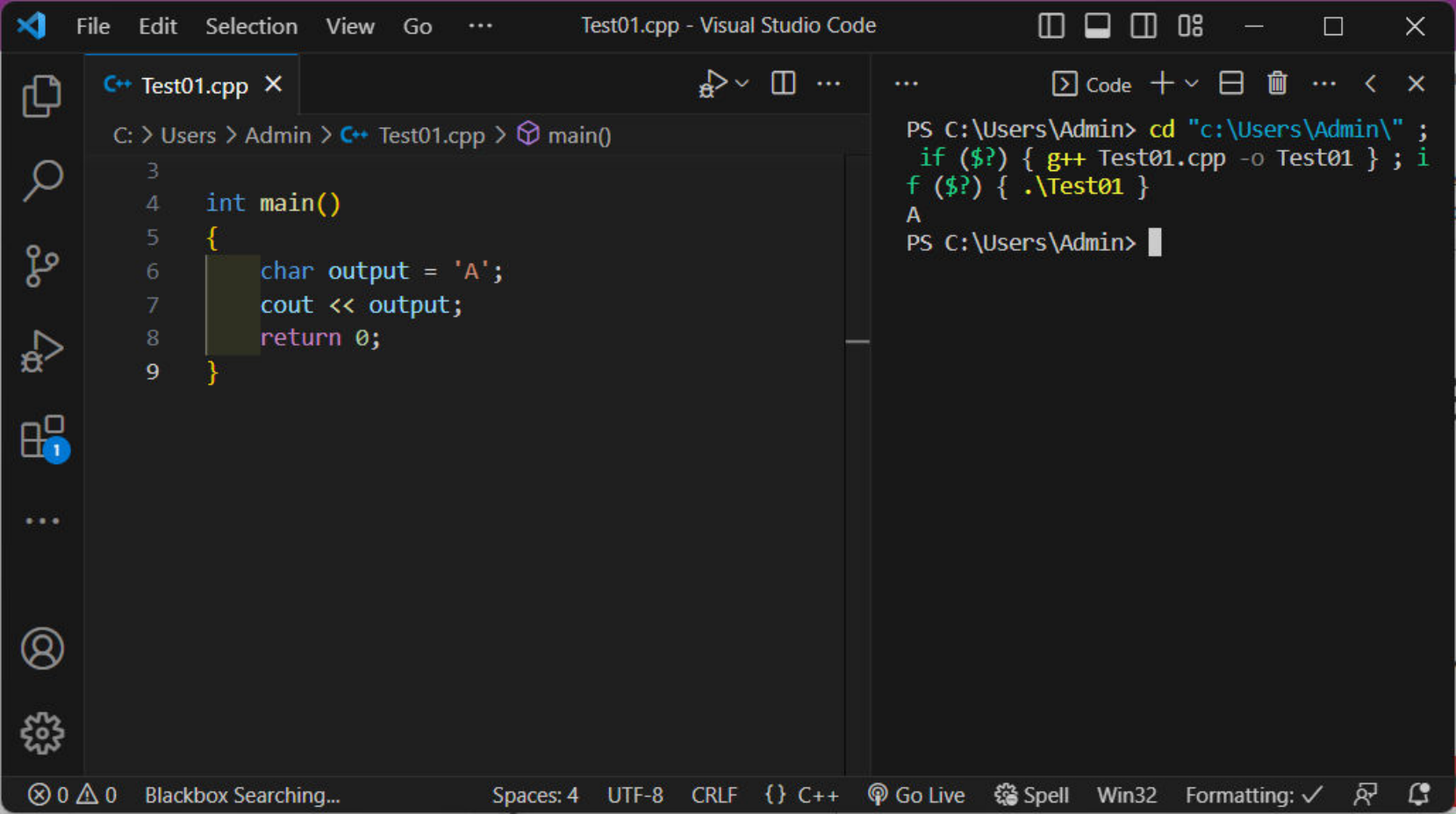
- It is possible to store int values in char variables

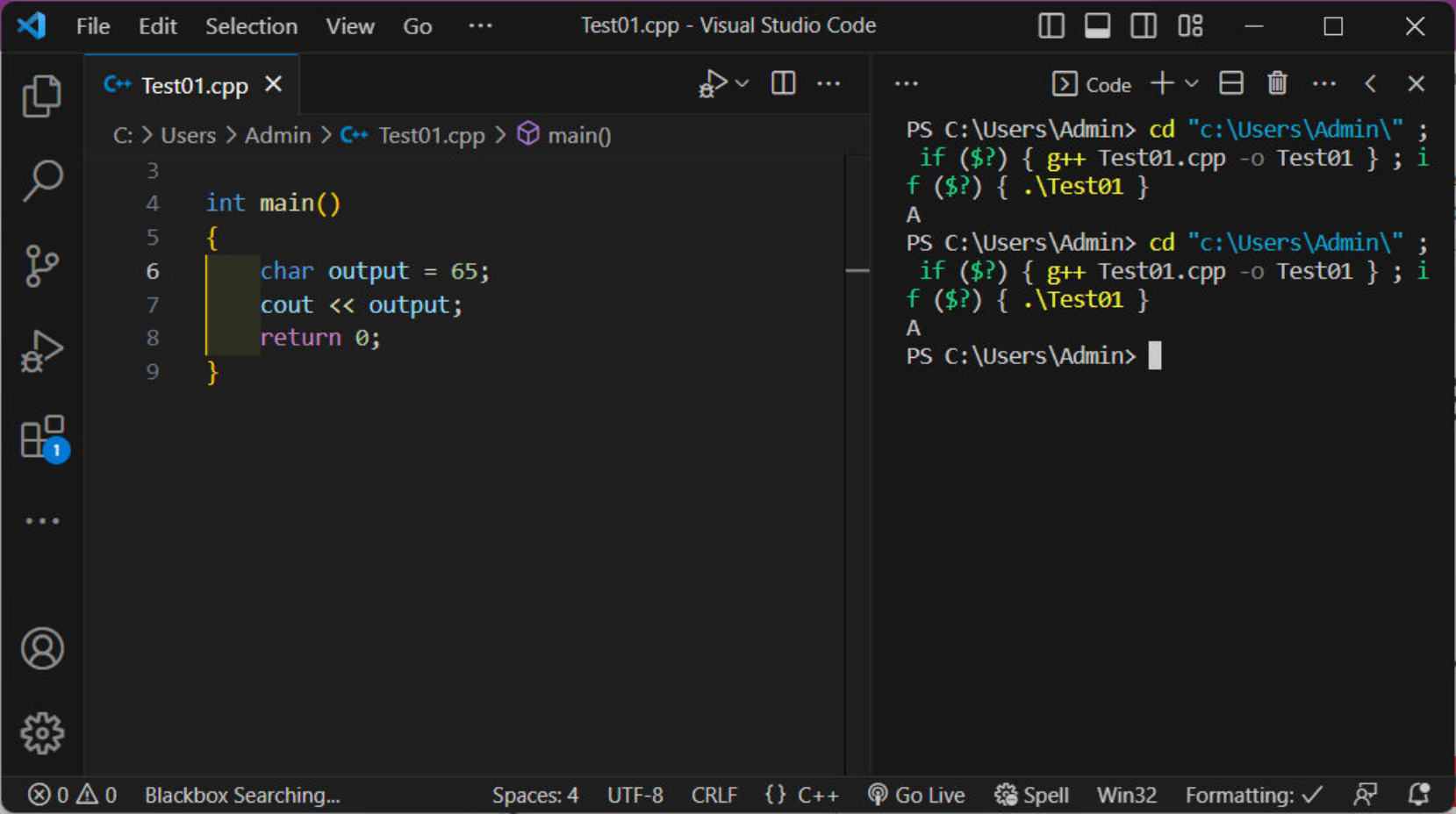
```
char letter = 65;
```

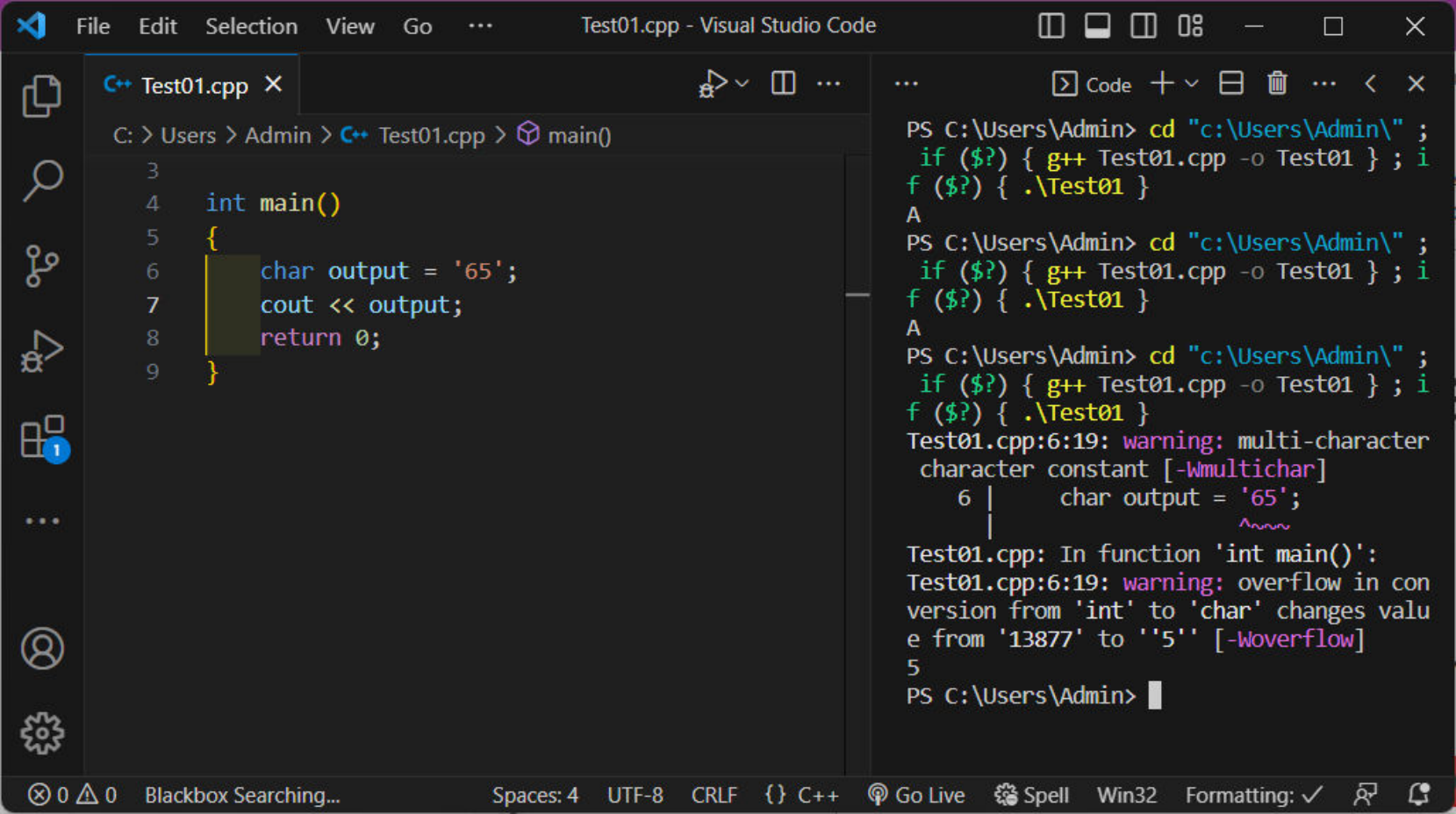

File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 int output = 65;
7 cout << output;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
65
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

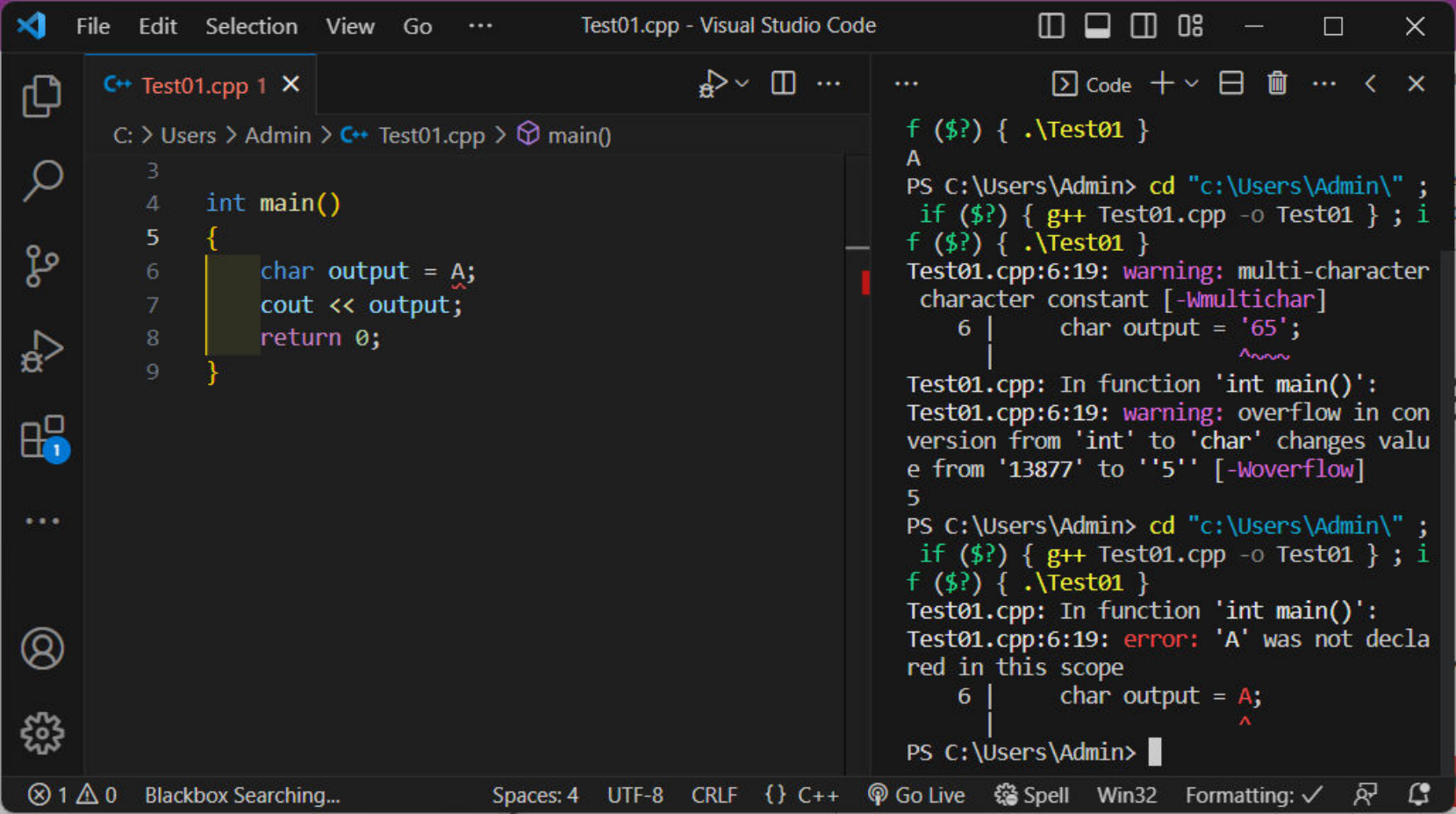

File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 int output = '65';
7 cout << output;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
65
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Test01.cpp:6:18: warning: multi-character character constant [-Wmultichar]
6 | int output = '65';
| ^~~~
13877
PS C:\Users\Admin>
0 0 Blackbox Searching... Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 int output = 'A';
7 cout << output;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
65
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Test01.cpp:6:18: warning: multi-character character constant [-Wmultichar]
6 | int output = '65';
| ^~~~
13877
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
65
PS C:\Users\Admin>
0 0 Blackbox Searching... Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 char output = 'A';
7 cout << output;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ;
if ($?) { g++ Test01.cpp -o Test01 } ; i
f ($?) { .\Test01 }
A
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 char output = 65;
7 cout << output;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
A
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
A
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 char output = '65';
7 cout << output;
8 return 0;
9 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
A
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
A
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Test01.cpp:6:19: warning: multi-character character constant [-Wmultichar]
6 | char output = '65';
| ^~~~
Test01.cpp: In function 'int main()':
Test01.cpp:6:19: warning: overflow in conversion from 'int' to 'char' changes value from '13877' to ''5'' [-Woverflow]
5
PS C:\Users\Admin>
0 0 Blackbox Searching... Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp 1
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 char output = A;
7 cout << output;
8 return 0;
9 }
Code
f ($?) { .\Test01 }
A
PS C:\Users\Admin> cd "c:\Users\Admin\" ;
if ($?) { g++ Test01.cpp -o Test01 } ; i
f ($?) { .\Test01 }
Test01.cpp:6:19: warning: multi-character
character constant [-Wmultichar]
6 | char output = '65';
| ^~~~
Test01.cpp: In function 'int main()':
Test01.cpp:6:19: warning: overflow in con
version from 'int' to 'char' changes valu
e from '13877' to ''5'' [-Woverflow]
5
PS C:\Users\Admin> cd "c:\Users\Admin\" ;
if ($?) { g++ Test01.cpp -o Test01 } ; i
f ($?) { .\Test01 }
Test01.cpp: In function 'int main()':
Test01.cpp:6:19: error: 'A' was not decla
red in this scope
6 | char output = A;
| ^
PS C:\Users\Admin>
1 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

bool ← → int

- The following actions are possible but generally not recommended!
- Values of type bool can be assigned to int variables
  - True is stored as 1
  - False is stored as 0
- Values of type int can be assigned to bool variables
  - Any non-zero integer is stored as true
  - Zero is stored as false

Arithmetic

- Arithmetic is performed with operators
  - + for addition
  - - for subtraction
  - * for multiplication
  - / for division

- Example: storing a product in the variable totalWeight

```
totalWeight = oneWeight * numberOfBars;
```

Results of Operators

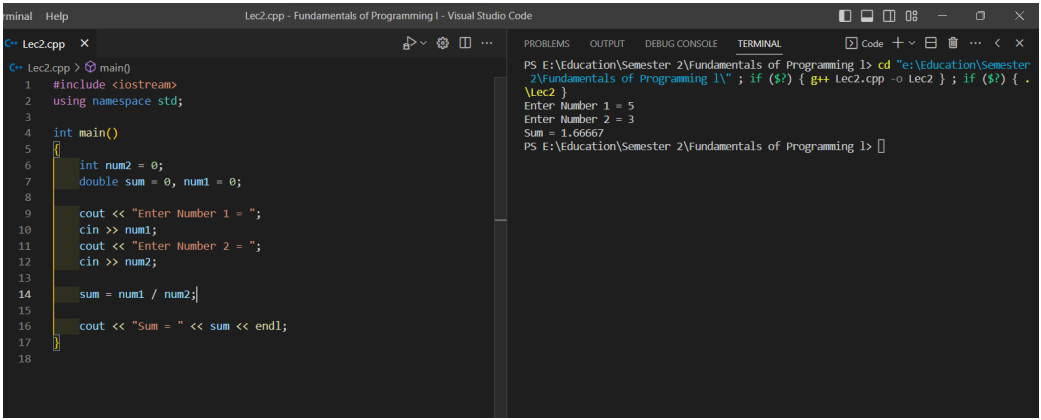
- Arithmetic operators can be used with any numeric type
- An operand is a number or variable used by the operator
- Result of an operator depends on the types of operands
  - If both operands are int, the result is int
  - If one or both operands are double, the result is double

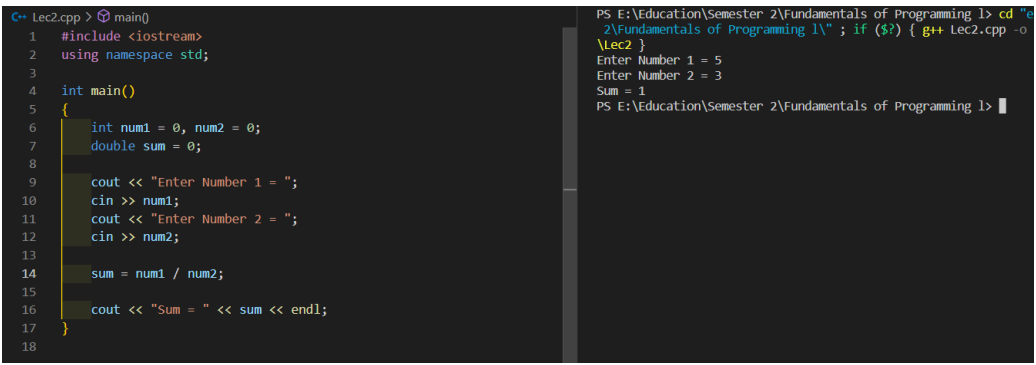
Division of Doubles

- Division with at least one operator of type double produces the expected results.

```
double divisor, dividend, quotient;
divisor = 3;
    dividend = 5;
quotient = dividend / divisor;
```

  - quotient = 1.6666…
!! - Result is the same if either dividend or divisor is of type int


rminal Help
Lec2.cpp - Fundamentals of Programming I - Visual Studio Code
Lec2.cpp
Lec2.cpp > main()
#include <iostream>
using namespace std;
int main()
{
int num2 = 0;
double sum = 0, num1 = 0;
cout << "Enter Number 1 = ";
cin >> num1;
cout << "Enter Number 2 = ";
cin >> num2;
sum = num1 / num2;
cout << "Sum = " << sum << endl;
}
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?) { g++ Lec2.cpp -o Lec2 } ; if ($?) { .\Lec2 }
Enter Number 1 = 5
Enter Number 2 = 3
Sum = 1.66667
PS E:\Education\Semester 2\Fundamentals of Programming 1>


Lec2.cpp > main()
#include <iostream>
using namespace std;
int main()
{
int num1 = 0, num2 = 0;
double sum = 0;
cout << "Enter Number 1 = ";
cin >> num1;
cout << "Enter Number 2 = ";
cin >> num2;
sum = num1 / num2;
cout << "Sum = " << sum << endl;
}
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?) { g++ Lec2.cpp -o \Lec2 }
Enter Number 1 = 5
Enter Number 2 = 3
Sum = 1
PS E:\Education\Semester 2\Fundamentals of Programming 1>

Division of Integers

- Be careful with the division operator!
  - int / int produces an integer result
    (true for variables or numeric constants)

```
int dividend, divisor, quotient;
dividend = 5;
divisor = 3;
quotient = dividend / divisor;
```

  - The value of quotient is 1, not 1.666…
!! - Integer division does not round the result, the fractional part is discarded!

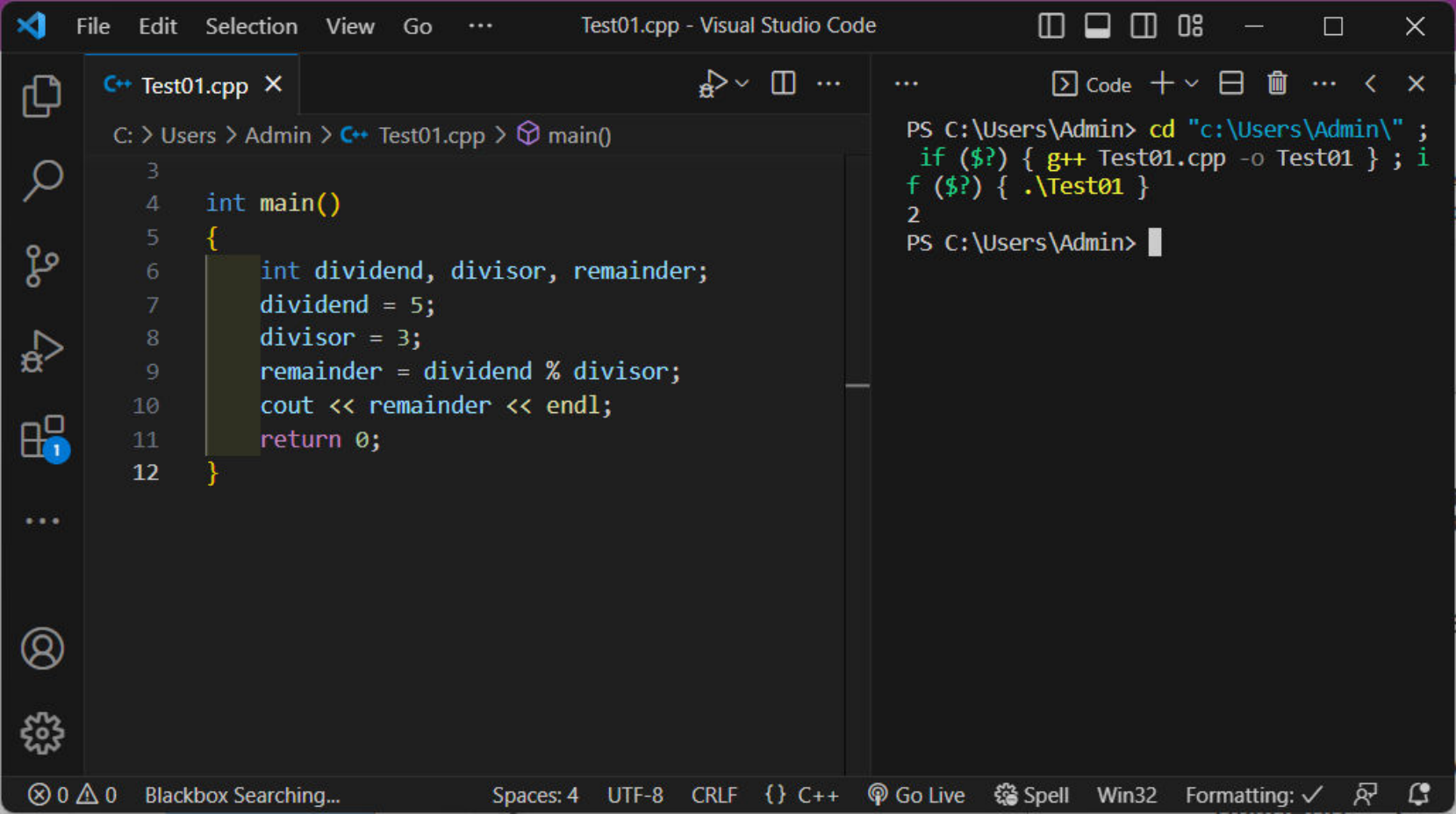
Integer Remainders

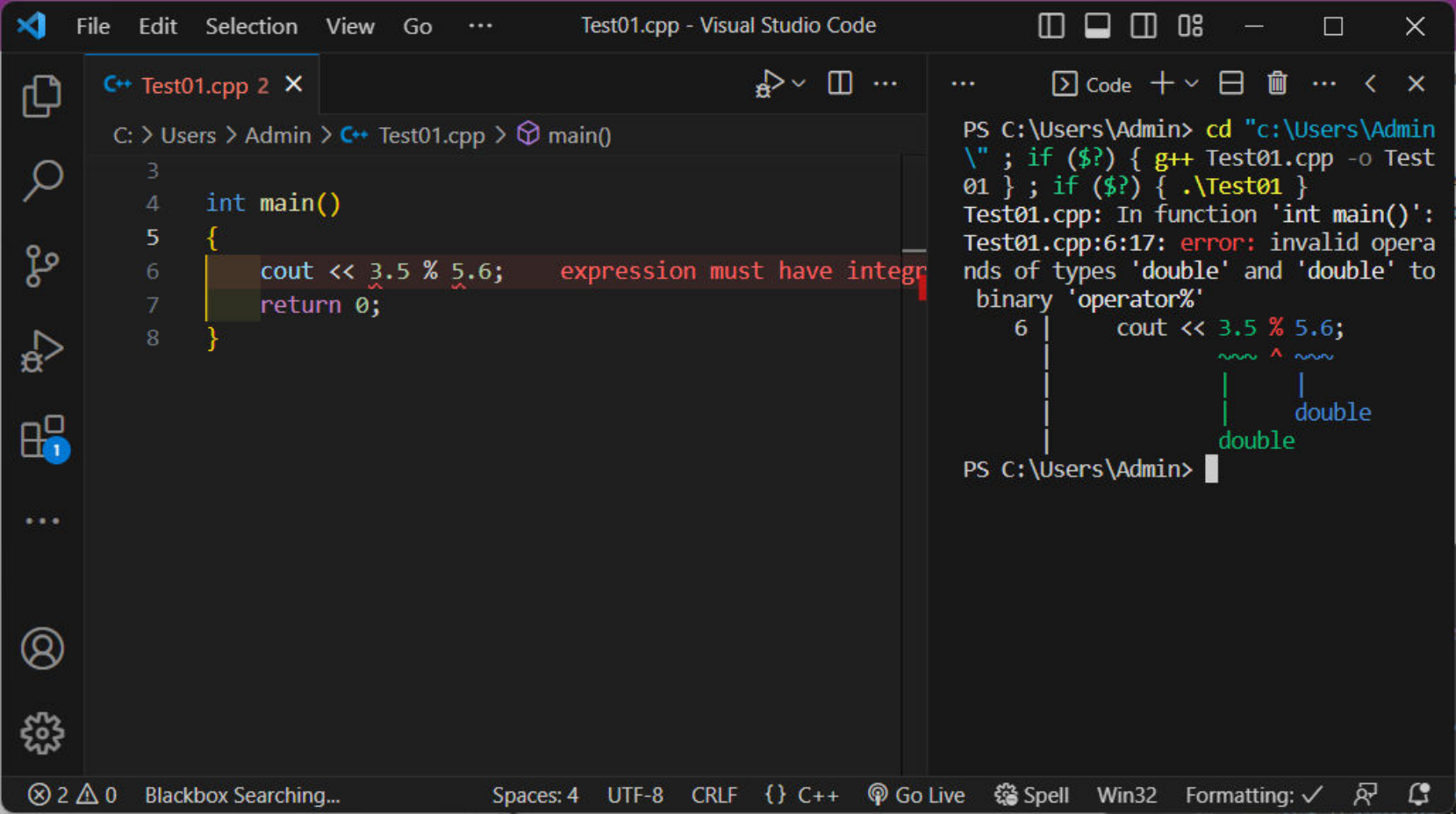
- % operator gives the remainder from integer division

- ```
  int dividend, divisor, remainder;
  dividend = 5;
  divisor = 3;
   remainder = dividend % divisor;
  ```

The value of remainder is 2

Display 2.6
  ```


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 int dividend, divisor, remainder;
7 dividend = 5;
8 divisor = 3;
9 remainder = dividend % divisor;
10 cout << remainder << endl;
11 return 0;
12 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
2
PS C:\Users\Admin>
0 0 Blackbox Searching...
Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go ···
Test01.cpp - Visual Studio Code
Test01.cpp 2
C: > Users > Admin > Test01.cpp > main()
3
4 int main()
5 {
6 cout << 3.5 % 5.6; expression must have integr
7 return 0;
8 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Test01.cpp: In function 'int main()':
Test01.cpp:6:17: error: invalid operands of types 'double' and 'double' to binary 'operator%'
6 | cout << 3.5 % 5.6;
| ~~~ ^ ~~~
| | |
| | double
| double
PS C:\Users\Admin>
2 0 Blackbox Searching... Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

Arithmetic Expressions

- Use spacing to make expressions readable
  - Which is easier to read?

Display 2.7

x+y*z or x + y * z

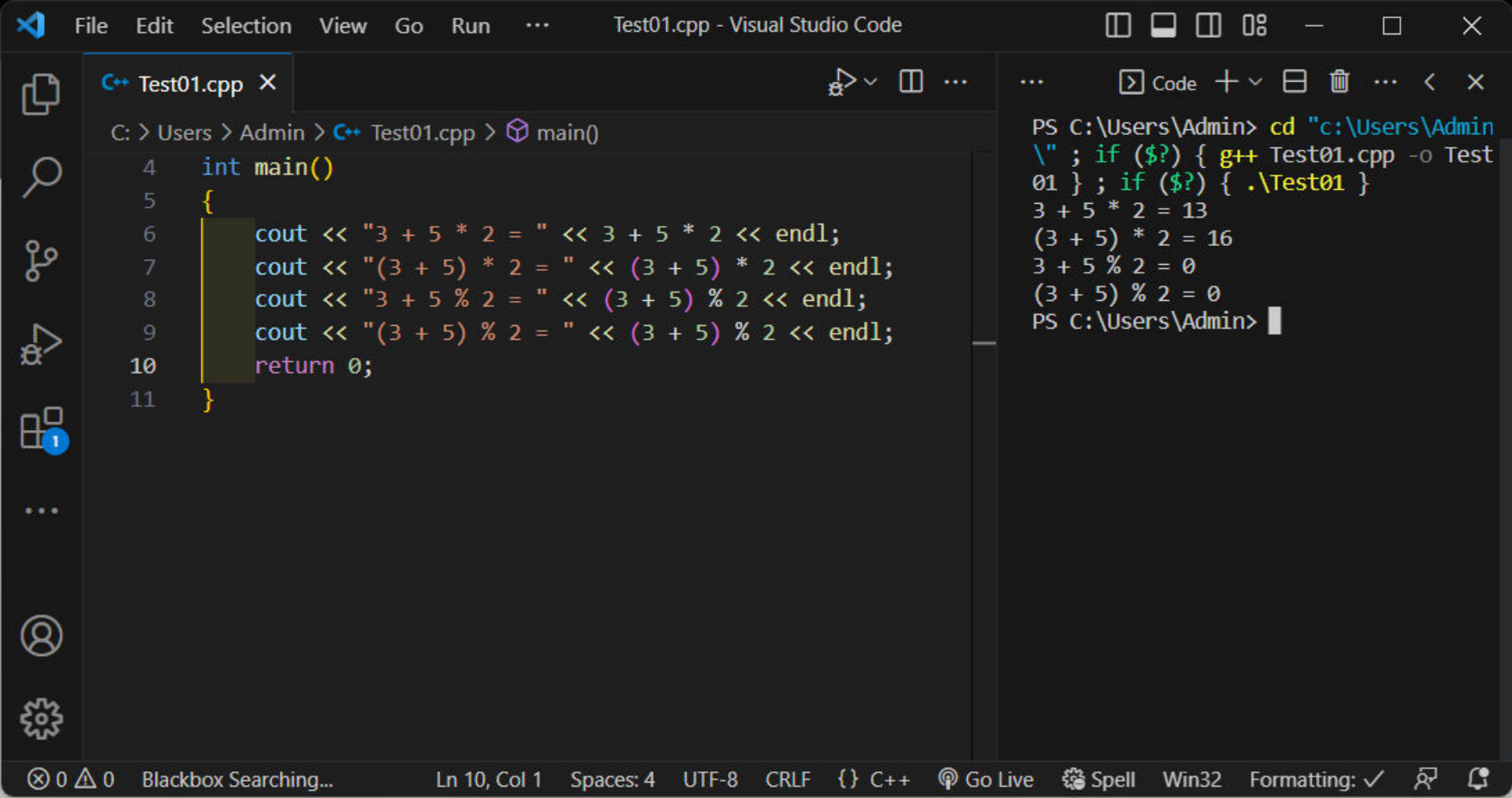
- Precedence rules for operators are the same as used in your algebra classes
- Use parentheses to alter the order of operations

  x + y * z (y is multiplied by z first)

  (x + y) * z (x and y are added first)

Operator Shorthand

- Some expressions occur so often that C++ contains to shorthand operators for them
- All arithmetic operators can be used this way
  - += count = count + 2; becomes
    count += 2;
  - *= bonus = bonus * 2; becomes
    bonus *= 2;
  - /= time = time / rushFactor; becomes
    time /= rushFactor;
  - %= remainder = remainder % (cnt1+ cnt2); becomes
    remainder %= (cnt1 + cnt2);


File Edit Selection View Go Run ··· Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
4 int main()
5 {
6 cout << "3 + 5 * 2 = " << 3 + 5 * 2 << endl;
7 cout << "(3 + 5) * 2 = " << (3 + 5) * 2 << endl;
8 cout << "3 + 5 % 2 = " << (3 + 5) % 2 << endl;
9 cout << "(3 + 5) % 2 = " << (3 + 5) % 2 << endl;
10 return 0;
11 }
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
3 + 5 * 2 = 13
(3 + 5) * 2 = 16
3 + 5 % 2 = 0
(3 + 5) % 2 = 0
PS C:\Users\Admin>
0 0 Blackbox Searching... Ln 10, Col 1 Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

Display 2.1 (1/2)

A C++ Program

```
#include <iostream>
 using namespace  std;
 int main( )
 {
    int numberOfBars;
    double oneWeight, totalWeight;

    cout << "Enter the number of candy bars in a package\n";
    cout << "and the weight in ounces of one candy bar.\n";
    cout << "Then press return.\n";
    cin >> numberOfBars;
    cin >> oneWeight;

    totalWeight = oneWeight * numberOfBars;

    cout << numberOfBars << " candy bars\n";
    cout << oneWeight << " ounces each\n";
    cout << "Total weight is " << totalWeight << " ounces.\n";

    cout << "Try another brand.\n";
    cout << "Enter the number of candy bars in a package\n";
    cout << "and the weight in ounces of one candy bar.\n";
    cout << "Then press return.\n";
    cin >> numberOfBars;
    cin >> oneWeight;

    totalWeight = oneWeight * numberOfBars;

    cout << numberOfBars << " candy bars\n";
    cout << oneWeight << " ounces each\n";
    cout << "Total weight is " << totalWeight << "  ounces.\n";

    cout << "Perhaps an apple would be healthier.\n";

     return 0;
}
```

Display 2.1
(2 /2)

A C++ Program (*part 2 of 2*)

Sample Dialogue

```
Enter the number of candy bars in a package
and the weight in ounces of one candy bar.
Then press return.
11 2.1
11 candy bars
2.1 ounces each
Total weight is 23.1 ounces.
Try another brand.
Enter the number of candy bars in a package
and the weight in ounces of one candy bar.
Then press return.
12 1.8
12 candy bars
1.8 ounces each
Total weight is 21.6 ounces.
Perhaps an apple would be healthier.
```

Display 2.2

DISPLAY 2.2 Some Number Types

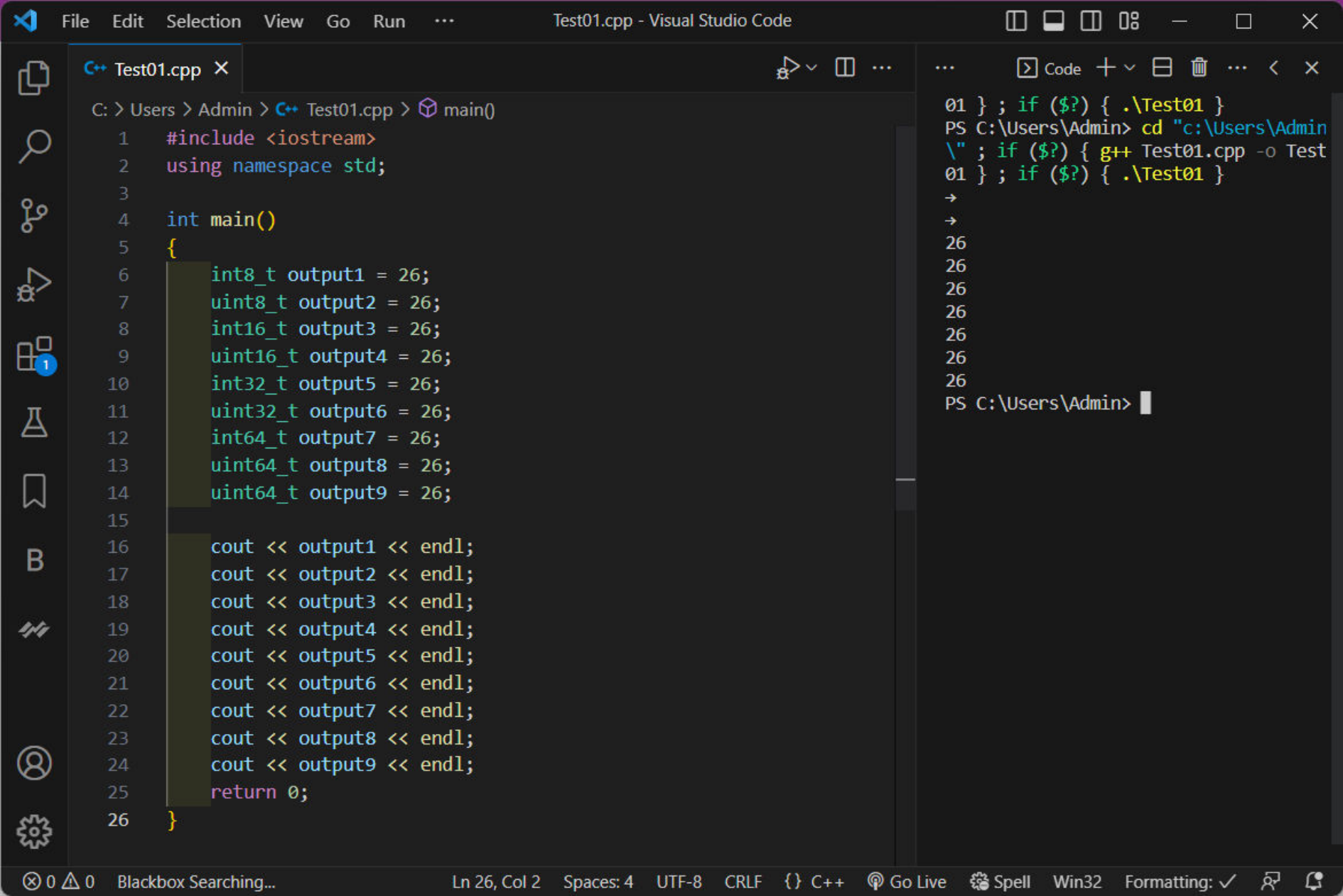
Type Name	Memory Used	Size Range	Precision
short (also called *short int*)	2 bytes	-32,768 to 32,767	(not applicable)
int	4 bytes	-2,147,483,648 to 2,147,483,647	(not applicable)
long (also called *long int*)	4 bytes	-2,147,483,648 to 2,147,483,647	(not applicable)
float	4 bytes	approximately 10^{-38} to 10^{38}	7 digits
double	8 bytes	approximately 10^{-308} to 10^{308}	15 digits
long double	10 bytes	approximately 10^{-4932} to 10^{4932}	19 digits

These are only sample values to give you a general idea of how the types differ. The values for any of these entries may be different on your system. **Precision** *refers to the number of meaningful digits, including digits in front of the decimal point. The ranges for the types* **float**, **double**, *and* **long double** *are the ranges for positive numbers. Negative numbers have a similar range, but with a negative sign in front of each number.*

Display 2.3

Some C++11 Fixed Width Integer Types

Type Name	Memory Used	Size Range
int8_t	1 bytes	-128 to 127
uint8_t	1 bytes	0 to 255
int16_t	2 bytes	-32,768 to 32,767
uint16_t	2 bytes	0 to 65,535
int32_t	4 bytes	-2,147,483,648 to 2,147,483,647
uint32_t	4 bytes	0 to 4,294,967,295
int64_t	8 bytes	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
uint64_t	8 bytes	0 to 18,446,744,073,709,551,615
long long	At least 8 bytes	


File Edit Selection View Go Run ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
#include <iostream>
using namespace std;
int main()
{
int8_t output1 = 26;
uint8_t output2 = 26;
int16_t output3 = 26;
uint16_t output4 = 26;
int32_t output5 = 26;
uint32_t output6 = 26;
int64_t output7 = 26;
uint64_t output8 = 26;
uint64_t output9 = 26;
cout << output1 << endl;
cout << output2 << endl;
cout << output3 << endl;
cout << output4 << endl;
cout << output5 << endl;
cout << output6 << endl;
cout << output7 << endl;
cout << output8 << endl;
cout << output9 << endl;
return 0;
}
Code
01 } ; if ($?) { .\Test01 }
PS C:\Users\Admin> cd "c:\Users\Admin
\" ; if ($?) { g++ Test01.cpp -o Test
01 } ; if ($?) { .\Test01 }
26
26
26
26
26
26
26
PS C:\Users\Admin>
0 0 Blackbox Searching...
Ln 26, Col 2 Spaces: 4 UTF-8 CRLF C++ Go Live Spell Win32 Formatting:

Display 2.4

The type *char*

```
#include <iostream>
using namespace std;
int main( )
{
    char symbol1, symbol2, symbol3;

    cout << "Enter two initials, without any periods:\n";
    cin >> symbol1 >> symbol2;

    cout << "The two initials are:\n";
    cout << symbol1 << symbol2 << endl;

    cout << "Once more with a space:\n";
    symbol3 = ' ';
    cout << symbol1 << symbol3 << symbol2 << endl;

    cout << "That's all.";

    return 0;
}
```

Sample Dialogue

```
Enter two initials, without any periods:
J B
The two initials are:
JB
Once more with a space:
J B
That's all.
```

Display 2.5

DISPLAY 2.5 The `string` Class

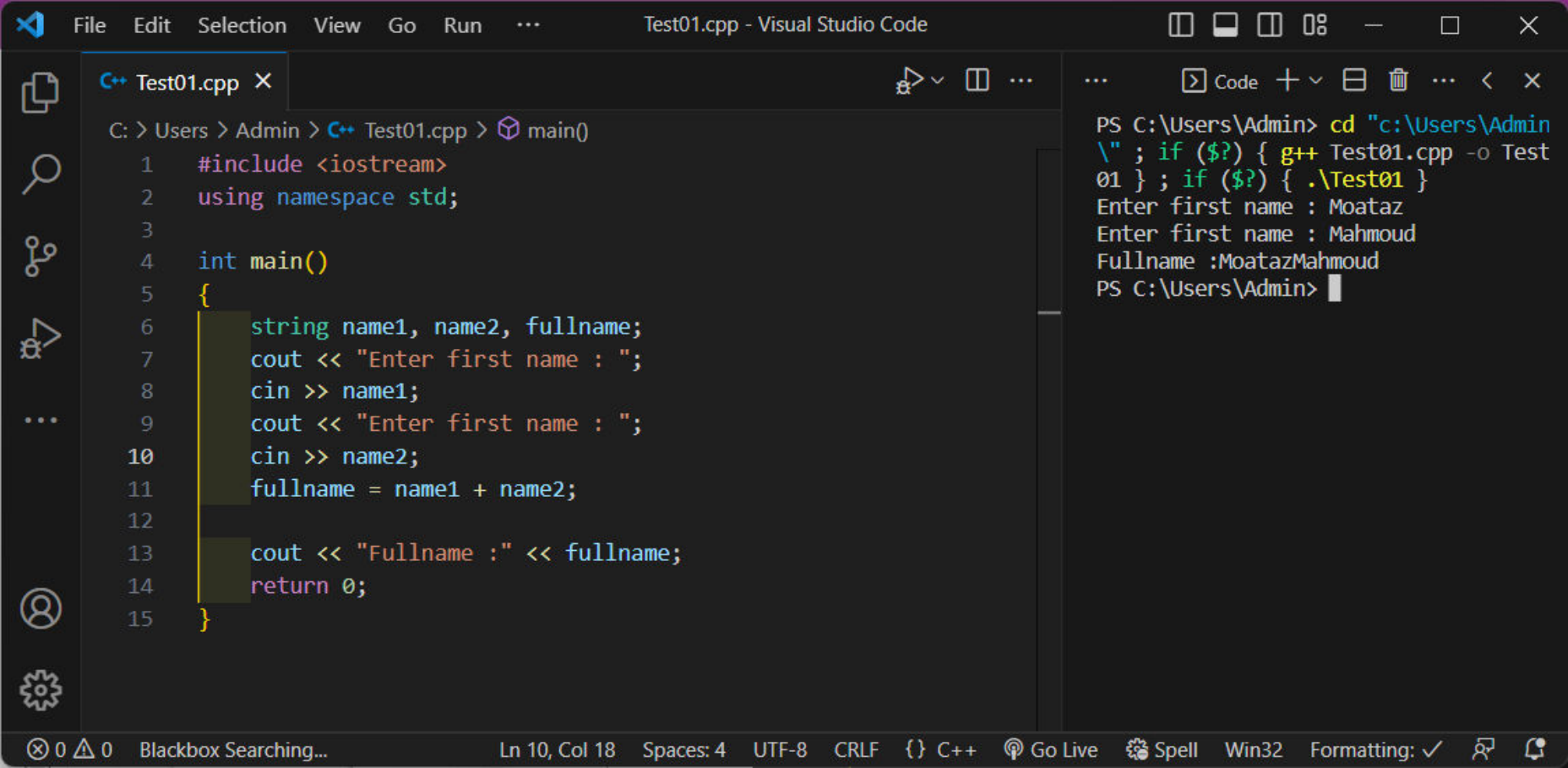
```
1     #include <iostream>
2     #include <string>
3     using namespace std;
4     int main()
5     {
6         string middleName, petName;
7         string alterEgoName;
8
9         cout << "Enter your middle name and the name of your pet.\n";
10        cin >> middleName;
11        cin >> petName;
12
13        alterEgoName = petName + " " + middleName;
14
15        cout << "The name of your alter ego is ";
16        cout << alterEgoName << "." << endl;
17
18        return 0;
19
20    }
```

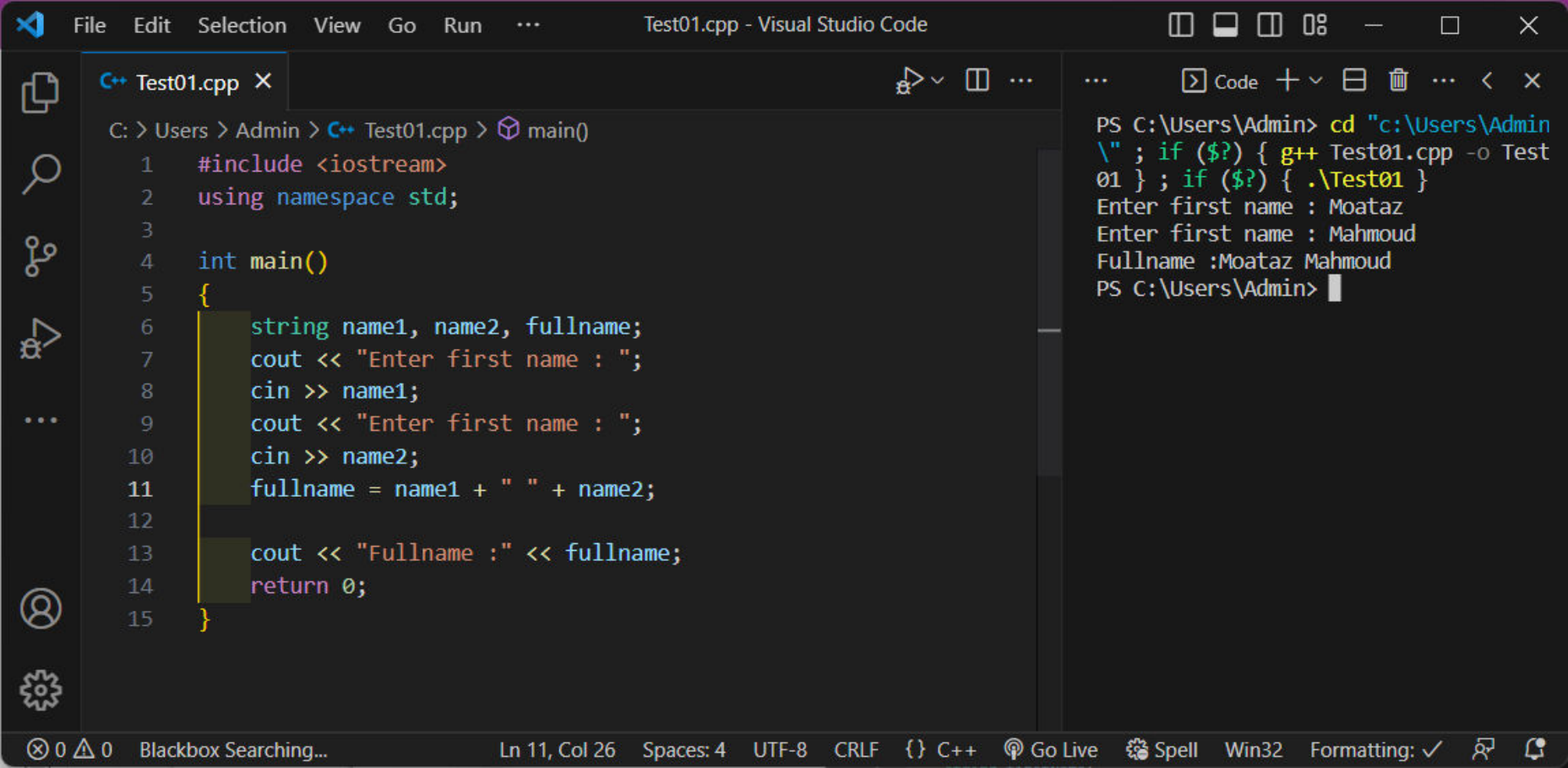
Sample Dialogue 1

```
Enter your middle name and the name of your pet.
Parker Pippen
The name of your alter ego is Pippen Parker.
```

Sample Dialogue 2

```
Enter your middle name and the name of your pet.
Parker
Mr. Bojangles
The name of your alter ego is Mr. Parker.
```


File Edit Selection View Go Run ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
#include <iostream>
using namespace std;
int main()
{
string name1, name2, fullname;
cout << "Enter first name : ";
cin >> name1;
cout << "Enter first name : ";
cin >> name2;
fullname = name1 + name2;
cout << "Fullname :" << fullname;
return 0;
}
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Enter first name : Moataz
Enter first name : Mahmoud
Fullname :MoatazMahmoud
PS C:\Users\Admin>
0 0 Blackbox Searching...
Ln 10, Col 18 Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


File Edit Selection View Go Run ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
#include <iostream>
using namespace std;
int main()
{
string name1, name2, fullname;
cout << "Enter first name : ";
cin >> name1;
cout << "Enter first name : ";
cin >> name2;
fullname = name1 + " " + name2;
cout << "Fullname :" << fullname;
return 0;
}
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Enter first name : Moataz
Enter first name : Mahmoud
Fullname :Moataz Mahmoud
PS C:\Users\Admin>
0 0 Blackbox Searching...
Ln 11, Col 26 Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

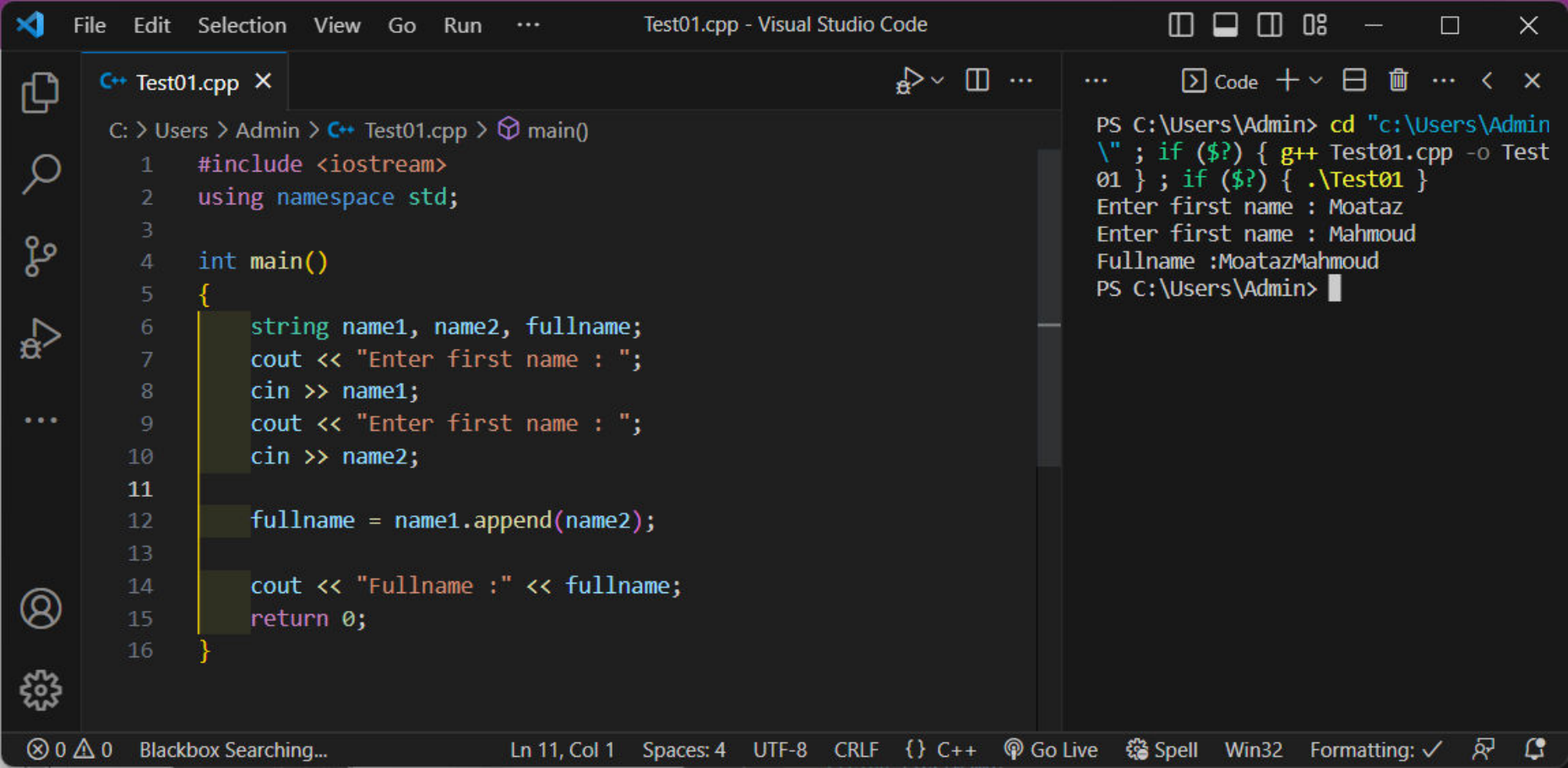
practice.cpp Untitled-1

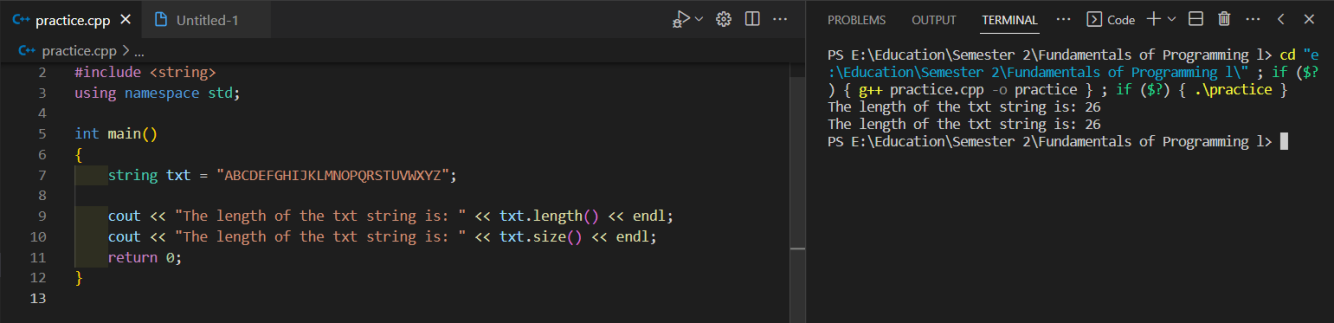
practice.cpp > ...

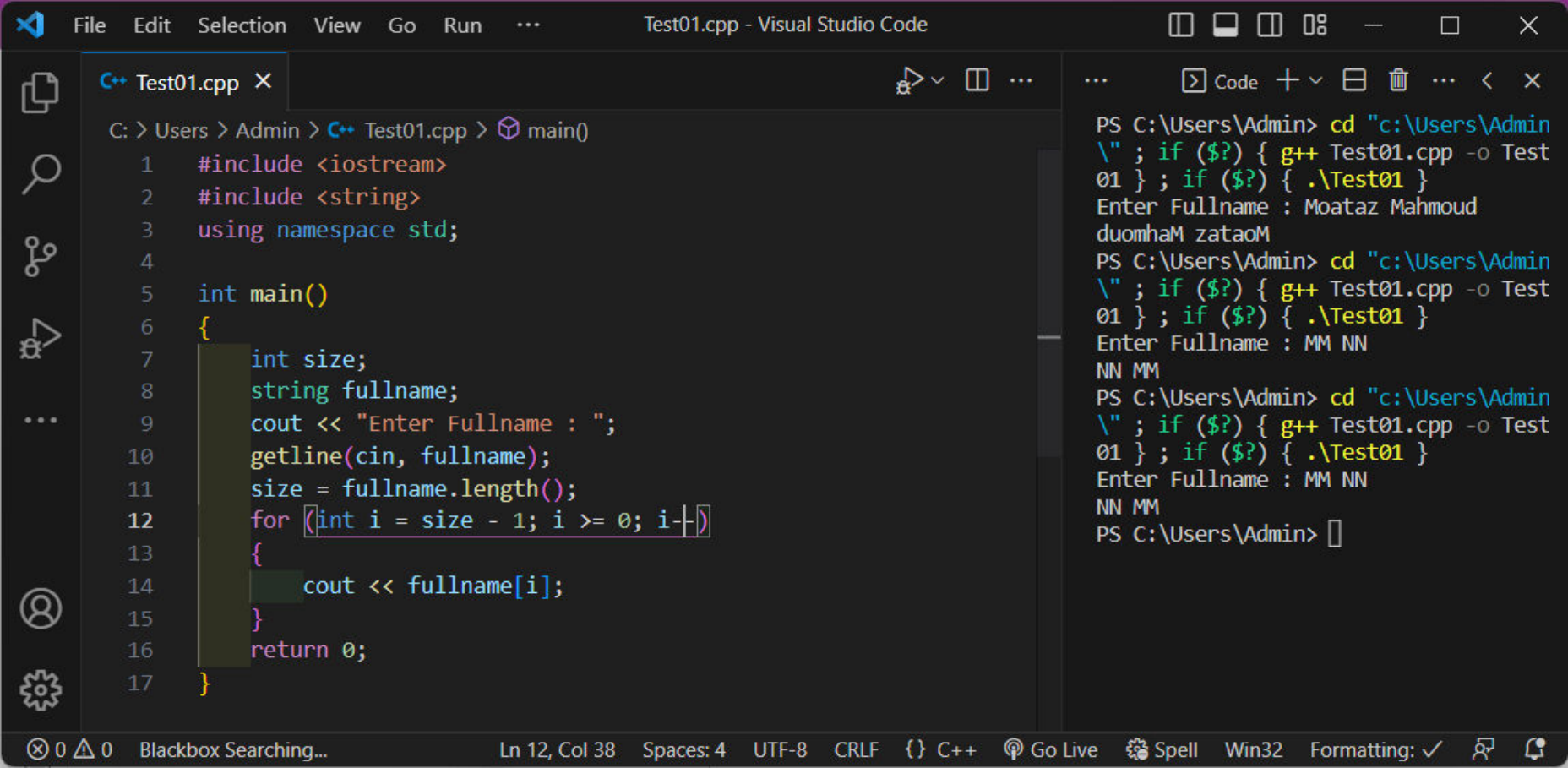
```cpp
#include <iostream>
#include <string>
using namespace std;

int main()
{
    string firstName;
    string lastName;
    string fullName;

    cin >> firstName;
    cin >> lastName;
    fullName = firstName.append(lastName);
    cout << fullName;
    return 0;
}

```

PROBLEMS OUTPUT TERMINAL ... Code

```
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
Moataz
Mahmoud
MoatazMahmoud
PS E:\Education\Semester 2\Fundamentals of Programming 1>
```


File Edit Selection View Go Run ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
#include <iostream>
using namespace std;
int main()
{
string name1, name2, fullname;
cout << "Enter first name : ";
cin >> name1;
cout << "Enter first name : ";
cin >> name2;
fullname = name1.append(name2);
cout << "Fullname :" << fullname;
return 0;
}
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Enter first name : Moataz
Enter first name : Mahmoud
Fullname :MoatazMahmoud
PS C:\Users\Admin>
0 0 Blackbox Searching...
Ln 11, Col 1 Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:


practice.cpp
Untitled-1
practice.cpp > ...
#include <string>
using namespace std;
int main()
{
string txt = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
cout << "The length of the txt string is: " << txt.length() << endl;
cout << "The length of the txt string is: " << txt.size() << endl;
return 0;
}
PROBLEMS
OUTPUT
TERMINAL
Code
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
The length of the txt string is: 26
The length of the txt string is: 26
PS E:\Education\Semester 2\Fundamentals of Programming 1>


File Edit Selection View Go Run ···
Test01.cpp - Visual Studio Code
Test01.cpp
C: > Users > Admin > Test01.cpp > main()
#include <iostream>
#include <string>
using namespace std;
int main()
{
int size;
string fullname;
cout << "Enter Fullname : ";
getline(cin, fullname);
size = fullname.length();
for (int i = size - 1; i >= 0; i--)
{
cout << fullname[i];
}
return 0;
}
Code
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Enter Fullname : Moataz Mahmoud
duomhaM zataoM
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Enter Fullname : MM NN
NN MM
PS C:\Users\Admin> cd "c:\Users\Admin\" ; if ($?) { g++ Test01.cpp -o Test01 } ; if ($?) { .\Test01 }
Enter Fullname : MM NN
NN MM
PS C:\Users\Admin>
0 0 Blackbox Searching...
Ln 12, Col 38 Spaces: 4 UTF-8 CRLF {} C++ Go Live Spell Win32 Formatting:

practice.cpp | Untitled-1

practice.cpp > main()

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

int main()
{
    string myString = "Hello";

    cout << myString[0];
    cout << myString[1];
    cout << myString[2];
    cout << myString[3];
    cout << myString[4];
    return 0;
}
```

PROBLEMS OUTPUT TERMINAL ... Code

```
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
Hello
PS E:\Education\Semester 2\Fundamentals of Programming 1>
```

practice.cpp | Untitled-1

practice.cpp > ...

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

int main()
{
    string myString = "Hello";

    for (int i = 0; i <= myString.size(); i++)
    {
        cout << myString[i];
    }

    return 0;
}
```

PROBLEMS OUTPUT TERMINAL ... Code

```
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
Hello
PS E:\Education\Semester 2\Fundamentals of Programming 1>
```

practice.cpp | Untitled-1

practice.cpp > main()

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

int main()
{
    string myString = "Hello";
    myString[0] = 'J';
    cout << myString;
    return 0;
}
```

PROBLEMS OUTPUT TERMINAL ... Code

```
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
Jello
PS E:\Education\Semester 2\Fundamentals of Programming 1>
```

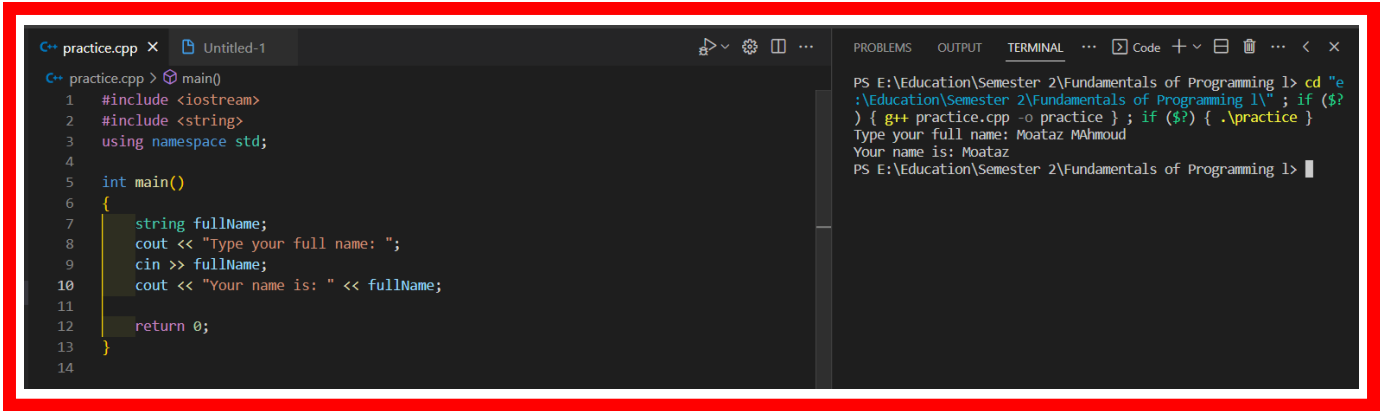
practice.cpp | Untitled-1

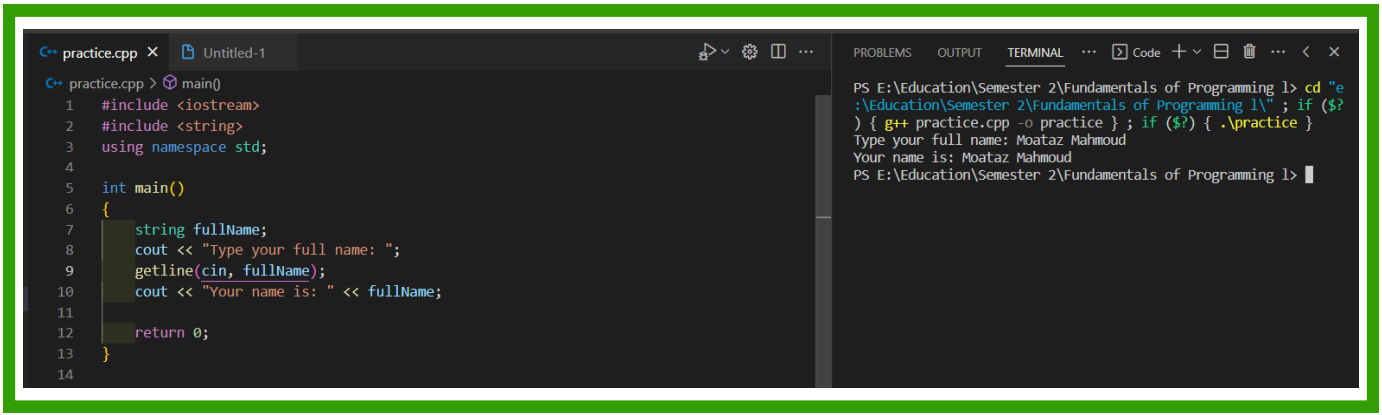
practice.cpp > main()

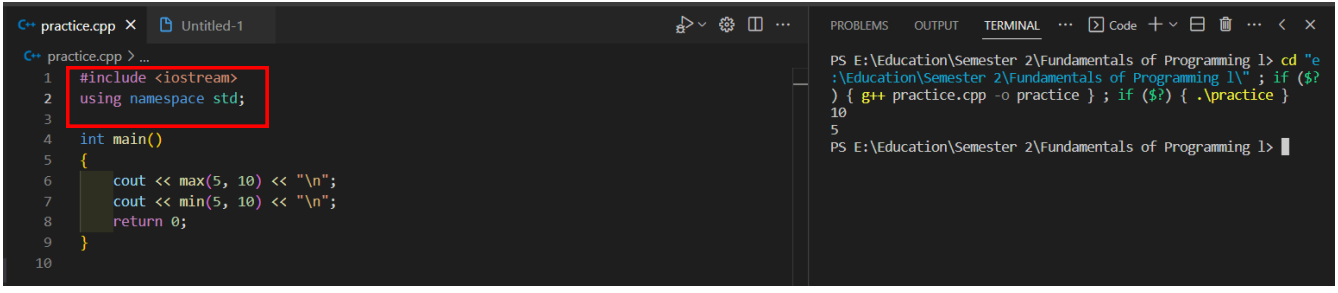
```cpp
#include <iostream>
#include <string>
using namespace std;

int main()
{
    string txt1 = "We are the so-called \"Vikings\" from the north.";
    string txt2 = "It\'s alright.";
    string txt3 = "The character \\ is called backslash.";

    cout << txt1 << "\n\n";
    cout << txt2 << "\n\n";
    cout << txt3 << "\n\n";

    return 0;
}
```

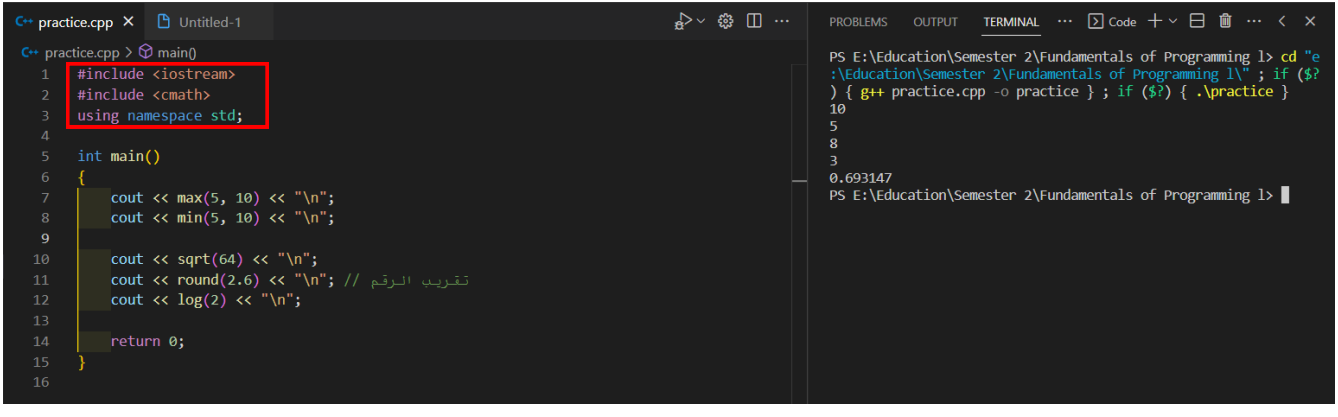
PROBLEMS OUTPUT TERMINAL ... Code

```
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
We are the so-called "Vikings" from the north.

It's alright.

The character \ is called backslash.

PS E:\Education\Semester 2\Fundamentals of Programming 1>
```


practice.cpp
Untitled-1
practice.cpp > main()
#include <iostream>
#include <string>
using namespace std;
int main()
{
string fullName;
cout << "Type your full name: ";
cin >> fullName;
cout << "Your name is: " << fullName;
return 0;
}
PROBLEMS
OUTPUT
TERMINAL
Code
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
Type your full name: Moataz MAhmoud
Your name is: Moataz
PS E:\Education\Semester 2\Fundamentals of Programming 1>


practice.cpp
Untitled-1
practice.cpp > main()
#include <iostream>
#include <string>
using namespace std;
int main()
{
string fullName;
cout << "Type your full name: ";
getline(cin, fullName);
cout << "Your name is: " << fullName;
return 0;
}
PROBLEMS
OUTPUT
TERMINAL
Code
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
Type your full name: Moataz Mahmoud
Your name is: Moataz Mahmoud
PS E:\Education\Semester 2\Fundamentals of Programming 1>


practice.cpp
Untitled-1
practice.cpp > ...
#include <iostream>
using namespace std;
int main()
{
cout << max(5, 10) << "\n";
cout << min(5, 10) << "\n";
return 0;
}
PROBLEMS
OUTPUT
TERMINAL
Code
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
10
5
PS E:\Education\Semester 2\Fundamentals of Programming 1>


practice.cpp
Untitled-1
practice.cpp > main()
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
cout << max(5, 10) << "\n";
cout << min(5, 10) << "\n";
cout << sqrt(64) << "\n";
cout << round(2.6) << "\n"; // تقريب الرقم
cout << log(2) << "\n";
return 0;
}
PROBLEMS
OUTPUT
TERMINAL
Code
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
10
5
8
3
0.693147
PS E:\Education\Semester 2\Fundamentals of Programming 1>

practice.cpp | Untitled-1

practice.cpp > main()

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

int main()
{
    cout << max(5, 10) << "\n";
    cout << min(5, 10) << "\n";

    return 0;
}
```

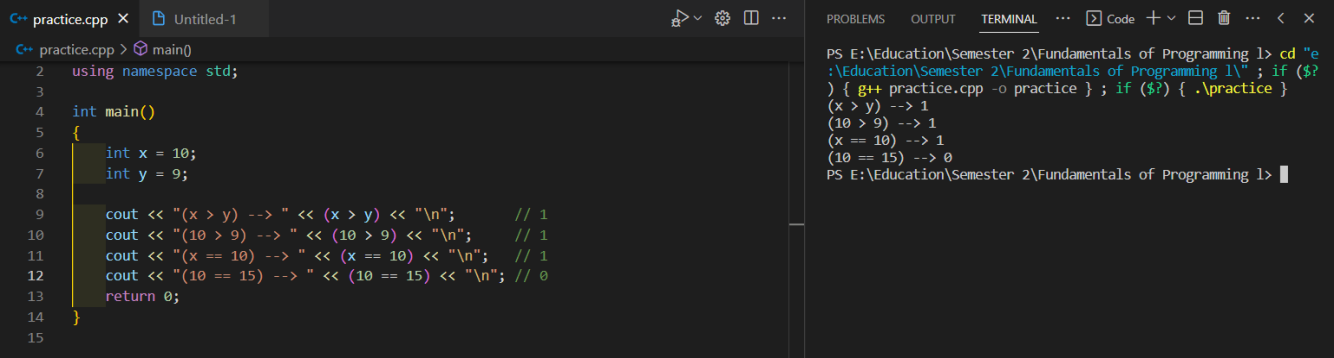
PROBLEMS OUTPUT TERMINAL Code

```
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e
:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?
) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
10
5
PS E:\Education\Semester 2\Fundamentals of Programming 1>
```

Other Math Functions

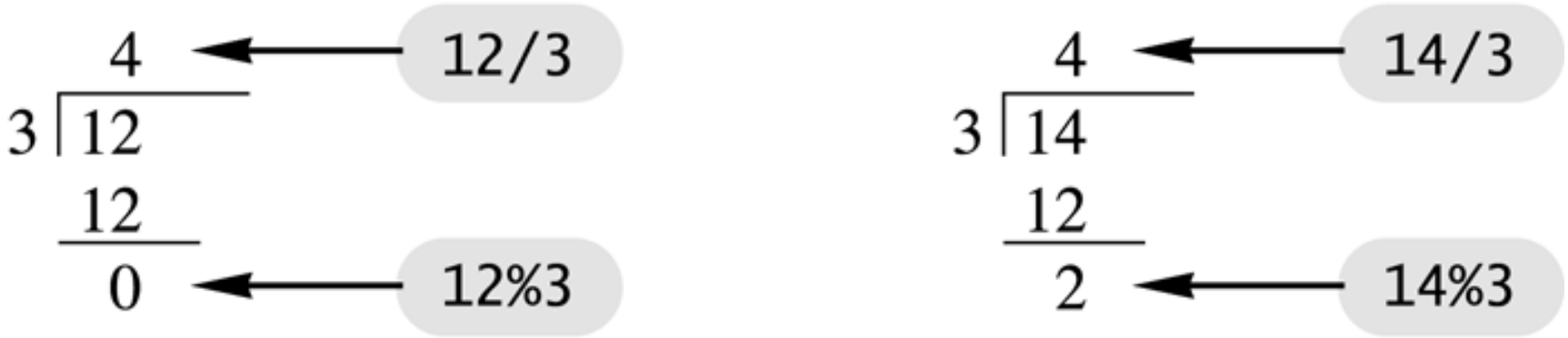
A list of other popular Math functions (from the `<cmath>` library) can be found in the table below:

Function	Description
abs(x)	Returns the absolute value of x
acos(x)	Returns the arccosine of x
asin(x)	Returns the arcsine of x
atan(x)	Returns the arctangent of x
cbrt(x)	Returns the cube root of x
ceil(x)	Returns the value of x rounded up to its nearest integer
cos(x)	Returns the cosine of x
cosh(x)	Returns the hyperbolic cosine of x
exp(x)	Returns the value of E^x
expm1(x)	Returns e^x -1
fabs(x)	Returns the absolute value of a floating x
fdim(x, y)	Returns the positive difference between x and y
floor(x)	Returns the value of x rounded down to its nearest integer
hypot(x, y)	Returns sqrt(x^2 +y^2) without intermediate overflow or underflow
fma(x, y, z)	Returns x*y+z without losing precision
fmax(x, y)	Returns the highest value of a floating x and y
fmin(x, y)	Returns the lowest value of a floating x and y
fmod(x, y)	Returns the floating point remainder of x/y
pow(x, y)	Returns the value of x to the power of y
sin(x)	Returns the sine of x (x is in radians)
sinh(x)	Returns the hyperbolic sine of a double value
tan(x)	Returns the tangent of an angle
tanh(x)	Returns the hyperbolic tangent of a double value


practice.cpp
Untitled-1
practice.cpp > main()
using namespace std;
int main()
{
int x = 10;
int y = 9;
cout << "(x > y) --> " << (x > y) << "\n"; // 1
cout << "(10 > 9) --> " << (10 > 9) << "\n"; // 1
cout << "(x == 10) --> " << (x == 10) << "\n"; // 1
cout << "(10 == 15) --> " << (10 == 15) << "\n"; // 0
return 0;
}
PROBLEMS
OUTPUT
TERMINAL
Code
PS E:\Education\Semester 2\Fundamentals of Programming 1> cd "e:\Education\Semester 2\Fundamentals of Programming 1\" ; if ($?) { g++ practice.cpp -o practice } ; if ($?) { .\practice }
(x > y) --> 1
(10 > 9) --> 1
(x == 10) --> 1
(10 == 15) --> 0
PS E:\Education\Semester 2\Fundamentals of Programming 1>

Display 2.6

Integer Division



Display 2.7

Arithmetic Expressions

Mathematical Formula	**C++ Expression**
$b^2 - 4ac$	`b*b - 4*a*c`
$x(y + z)$	`x*(y + z)`
$\frac{1}{x^2 + x + 3}$	`1/(x*x + x + 3)`
$\frac{a + b}{c - d}$	`(a + b)/(c - d)`