Week 7 cs150

Share

★ Leave the first rating

Terms in this set (102)

The return value of the getline() function is a string object.	false
The cout object is an instance of the ofstream class.	false

the cin object is an instance of the istream class	true
The line: cin >> n; throws a runtime exception if n is an int and it tries to read the input "one".	false
A catch block specifies the type of exception it can catch and immediately terminates the program.	false
The order of the catch blocks does not affect the program	false
If the catch block with an ellipses (in the heading) is needed, then it should be the first catch block in a sequence of try/catch blocks.	false
One of the main problems with the completion code strategy of error handling is that callers can ignore the return value without encountering any warnings.	true
A completion code is a special return value that means "the function failed to execute correctly."	true
The C++ term for what is called a ***subclass*** in other languages is ***derived*** class.	true
In the ***primed loop pattern***, you read data before the loop and at the end of the loop.	True
In the ***primed loop pattern***, you use a break statement to exit the loop when the sentinel is found.	False

In the primed loop pattern, you use Boolean flag to signal when the sentinel is found.	false
In the ***loop-and-a-half***, you use a break statement to exit the loop when the sentinel is found.	True
In the ***loop-and-a-half pattern***, you read data before the loop and at the end of the loop.	False
In the flag-controlled-pattern, you read data before the loop and at the end of the loop.	false
The predefined constant _cpluplus indicates which version of the C++ standard is being used.	True
The standard library version of sqrt(-2) returns the not-a-number error code.	true
The standard library version of sqrt(-2) throws a runtime exception because there is no possible answer.	False
If no exception is thrown in a try block, all catch blocks associated with that try block are ignored.	true
You can report a logical error encountered in your code by using the throw keyword.	true
The heading of a try block can contain ellipses in place of a parameter.	False

If an output stream's file is missing when you try to open it, its fail() member function returns true.	false

To read a line of text, you include the header file <string>.	true
To use strings as a data stream source or sink, use the <sstream> header	true
To use a disk file as a data stream source or sink, use the <fstream> header	true
To use a disk file as a data stream source or sink, use the <ofstream> header	False
To use strings as a data stream source or sink, use the <stringstream> header	false
Formatted I/O means that you read and write data token-by-token	True
When writing a function with stream parameters, always use the **most general** type of stream that meets the specification.	True
The preprocessor operates on code **before** it has been compiled (ie. while it is still at the textual level)	True

The cout object is an instance of the ostream class	true
The preprocessor operates on code **after** it has been compiled.	false
The statement #if abs(-3) > 2 is legal.	false
The directives #if defined(symbol) and #ifndef symbol mean, essentially, the same thing.	false
Unformatted I/O means that you read and write data character-by-character.	True
The #if preprocessor directive may compare double literals but not variables.	false
To use different versions of a function depending on the platform is called conditional compilation.	true
What happens when this code fragment runs?istringstream in**(**".5"**);** **int** n**;** in **>>** n**;**	it sets an error state in in.
What happens when this code fragment runs? istringstream in("12.5"); int n; in >> n;	n is set to 12
What happens when you execute the (erroneous) line: auto x = sqrt(-1);	The function returns an error value and the program continues

What happens when you execute the (erroneous) line: assert(2 + 2 == 5);	The program prints an error message and terminates
What happens when you execute the following (erroneous) code: istreamstring in("one"); int n; in >> n;	The stream is put into a failed state, but the program continues running
What happens when you execute the (erroneous) line: bool ok = 2 + 2 == 5;	Nothing happens. The variable ok is set to false.
Code that may cause an error should be placed in a _____ block and code that handles the error should be inside a _____ block?	try, catch

What is correct for # 1-4?
int main()
{
{ string s= "hello";
cout <<s.at(5) <<endl;}
#2 (#3) {
cout << e. #4() << endl;}}
	1)try
	2)catch
	3)exception&
	4)what

What happens when you execute the following (erroneous) code: cout << stoi(42.5) << endl;	The code does not compile because the argument is the wrong type
What happens when this code fragment runs in C++ 11? cout << stoi("one") << endl;	it throws a runtime exception
What happens when this code fragment runs?	stoi() returns 12

Assume the user types "brown cow" when this code runs. What prints? **int** n**;** **if (**cin **>>** n**)** cout **<<** "X\n"**;** **else** cout **<<** "Y\n"**;**	Y

The file expenses.txt contains the line: Hotel, 3 nights. $ 1,750.25. What prints?
ifstream in("expenses.txt");
char c;
while (in.get(c)){
if (isdigit(c)) {
in.unget(); i
nt n; in >> n;
cout << n << 'x'; }}
	3x1x750x.25x

The file grades.txt contains lines of text that look like this: Smith 94 Jones 75 . . . Each line of text contains the student's name (a single word) and an integer score. What is the legal way of reading one student's information, given the following code? string name; int score; ifstream in("grades.txt");	in >> name >> score;

What prints?
string s("hello");
try {
if (s.size() > 2) throw 42;
if (islower(s.back())) throw "goodbye";
if (s == "hello") throw string("hello");
s.at(s.size()) = 'x';
cout << "one\n";}
catch (const int& e) { cout << "two\n"; }
catch (const string& e) { cout << "three\n"; }
catch (exception& e) { cout << "four\n"; }
catch (...) { cout << "five\n"; }
	two

What prints?
string s("hello");
try {
auto x = s.at(s.size());
cout << "one" << endl;}
catch (const string& e) { cout << "two\n"; }
catch (exception& e) { cout << "three\n"; }
catch (...) { cout << "four\n"; }
	three

What happens when this code fragment runs in C++ 11? cout **<<** sqrt**(-**2**) <<** endl**;**	sqrt() returns a not-a-number error value
Assume that the file scores.txt does not exist. What happens? ofstream out**(**"scores.txt"**);** out **<<** "Peter" **<<** " " **<<** 20 **<<** endl**;** out **<<** "John" **<<** " " **<<** 50 **<<** endl**;**	Creates a new file, scores.txt and writes two lines of text.
Which of the following loop patterns are used here? **int** upper **=** 0**;** **char** ch**;** **while (**in.get**(**ch**)){** **if (**ch **>=** 'A' **&&** ch **<=** 'Z'**)** upper**++;}**	-data loop -inline test
Which of the following loop patterns are used here? **int** n**;** in **>>** n**;** **while (**abs**(**n**)){** out **<<** n **%** 4 **<<** endl**;** n **/=** 4**;}**	limit loop
Which of the following loop patterns are used here? string s{"hello CS 150"}; for (auto e : s){ if (toupper(e)) break;}	-loop and a half -iterator or range loop
Which of the following loop patterns are used here? string s{"hello CS 150"}; for (auto e : s){ if (toupper(e)) out.put('x');}	iterator or range loop

The file temp.txt contains "Orange Coast College". What prints?
ifstream in("temp.txt");
char c;
while (in.get(c)){
if (isupper(c)) cout << toupper(c);}
	OCC

The file temp.txt contains "If I saw an Aardvark, I would scream!". What prints?
`ifstream in("temp.txt");`
`char c;`
`int i = 0;`
`while (in.get(c)){`
`if (tolower(c) == 'a')`
`i++;}`
`cout << i << endl;`
	6

What does this code do?
`ifstream in("temp.txt");`
`string x;`
`int i{0};`
`while (in >> x) i++;`
`cout << i << endl;`
	Counts the number of words in the file

What does this code do?
`ifstream in("temp.txt");`
`char x;`
`int i{0};`
`while (in.get(x)) i++;`
`cout << i << endl;`
	Counts the number of characters in the file

What does this code do?
`ifstream in("temp.txt");`
`char x;`
`int i{0};`
`while (in >> x) i++;`
`cout << i << endl;`
	Counts the number of non-space characters in the file

In the C++ stream hierarchy, base class of the istream class is:	ios
In the C++ stream hierarchy, the base class of the ostream class is:	ios
In the C++ stream hierarchy, the base class of the fstream class is:	iostream
In the C++ stream hierarchy, the base class of the ofstream class is:	ostream

What term describes this block of code?
`#if _APPLE_`
`istringstream in(" .75");`
`int n = 3;in >> n;`
`#endif`
	conditional compilation

`if (_APPLE_)`
`cout << "Running on a Mac" << endl;`
`else if (_WIN32)`
`cout << "Running on Windows" << endl;`
`else if (_linux)`
`cout << "Running on Linux" << endl;`
`else`
`cout << "Running on an unknown platform" << endl;`
	The program will not compile

`#if _APPLE_`
`cout << "Running on a Mac" << endl;`
`#elif _WIN32`
`cout << "Running on Windows" << endl;`
`#elif _linux cout << "Running on Linux" << endl;`
`#else cout << "Running on an unknown platform" << endl;`
`#endif`
	Only the lines that identify your platform will be included in the executable

This loop:
`string str;`
`while (getline(in, str)){ cout << str << endl;}`
	illustrates line-based stream processing

/What happens when this code fragment compiles and runs?
`#define N`
`#ifdef N`
`cout << "Hello";`
`#else cout << "Goodbye";`
`#endif`
	prints "Hello"

What happens when this code fragment compiles and runs?
`#define N`
`#ifndef N // versus ifdef`
`cout << "Hello";`
`#else`
`cout << "Goodbye";`
`#endif`
	prints "Goodbye"

After writing data to an ostringstream object named os, you can retrieve the string it contains by using:	os.str()
Stream arguments to a function should always be passed:	by reference
The line: ifstream in("x"); throws a runtime exception if a file x cannot be found	false

Which line opens the file in.txt for reading?	ifstream in("in.txt");
Suppose you have written a non-interactive program that inputs data from a file. If the input file does not exist when the program executes, then you should choose which option?	Terminate the program with an error message
indefinite limit loop that reduces its input	while (n!=0) {n/=2;}
indefinite limit loop that uses successive approximations	while (abs(g1-g2)) >= EPSILON {...}
indefinite data loop that uses formatted input	while (cin >>n){...}
indefinite data loop that uses raw inpt	while(cin.get(ch)){...}
counter-controlled symmetric loop for producing a sequence of data	for (inti=12; <=19; ++i)
counter controlled asymmetric loop for processing characters	for(size_t i=0; len=s.size(); i<len; i++) {..}
counter-controlled loop for processing substrings	for (size_t i=4, slen=4, len=s.size(); i<len; i++)
iterator loop that may change its container	for (auto&e: col) {...}
iterator loop that cannot change its container	for (auto e:col){...}
read and write characters to memory using streams	sstream
connect a disk file to an input or output stream	fstream
used the predefined stream objects cin and cout	iostream
determines the category of a character	cctype
modify the way that memory is converted to characters on input or output	iomanip
What statement is used to signal other parts for your program that a particular error has occurred?	throw
What header file to you need to include to use the standard C++ error-handling classes?	<stdexcept>
A(n) _____ is a statement about a condition which must be true when it is encountered in your code.	assertion
Programmers use _____ to reason about logical correctness in their code.	assertions
After opening the input stream in, which of these cannot be used to see if the file was successfully opened?	if (in.opened()){/ **opened ok**/}
What preprocessor directive is not used when you wish to create blocks of code that are only compiled under certain circumstances? #ifdef #define #if #ifndef	All of these may be used