midterm 3 cs150

Share

11 studiers today ★ Leave the first rating

Terms in this set (111)

An _______ is an object which specifies the position of an element inside a container, regardless of what kind of container you use. pointer subscript index iterator lambda	...
What is x? vector<int> v{1, 2, 3}; auto x = min_max_element(v.begin(), v.end()); 1 an iterator a std::pair object None of these 3	...
The C++11 standard library provides the function stoi() to convert a string to an integer. Which library is it found in? None of these iostream cmath cnvt string	string
What happens when this code fragment compiles and runs? #define N #ifndef N cout << "Hello";#else cout << "Goodbye"; #endif It does not compile. prints "Goodbye" prints "HelloGoodbye" prints "Hello" prints nothing	prints "Goodbye"

struct Employee
{
long empID;
std::string lastName;
double salary;
int age;
};
Employee bob{777, "Zimmerman"};
 	
None of these
salary
empID
lastName
age	

An unnamed (anonymous) function is called a(n): iterator None of these functor lambda stub	lambda

Assuming that you have an iterator named iter. Which statement retrieves the element that iter refers to?
 	
auto c = &iter;
auto a = ++iter;
auto b = *iter;
auto d = iter;
None of these
	auto b = *iter;

Assume that ppi correctly points to pi. Which line prints the address of ppi?
int main()
{
double pi = 3.14159;
double *ppi;
// code goes here
// code goes here
}
 	
None of these
cout << &pi;
cout << &ppi;
cout << *ppi;
cout << ppi;
	cout << &ppi;

What is the address of the first pixel in the last row of this image?
Pixel *p; // address of pixel data
int w, h; // width and height of image
 	
p + w * h
p + w + (h - 1)
p + w + h
None of these are correct
p + w * (h - 1)
	p + w * (h - 1)

int *p;
*p = n;
cout << *p << endl;

None of these
Will not compile
No compilation errors, but undefined behavior when run
The value 0 (stored in n)
The address value where n is stored

What happens with the following section of code?
if (_APPLE_)cout << "Running on a Mac" << endl;
else if (_WIN32)cout << "Running on Windows" << endl;
else if (_linux)cout << "Running on Linux" << endl;
elsecout << "Running on an unknown platform" << endl;

the program will not compile
All lines will be included in the program.
It will print the platform you are running on.
The program will crash if compiled on one platform, but run on another.
Only the lines that identify your platform will be included in the executable

All lines will be included in the program. It will print the platform you are running on. wrong
Only the lines that identify your platform will be included in the executable wrong

---

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

conditional compilation
proprietary compilation
None of these
selection statements
alternative compilation
compiler directives

conditional compilation

In a sequence of try/catch blocks, the last catch block of that sequence should be ___.

catch(exception){}
catch(int x){ }
catch(str){ }
catch(...){ }
PreviousNext

catch(. ..){ }

The logic_error and runtime_error classes are defined in the header file ___.

exception
stdlib
stdexcept
stdex

stdexcept

int n; in >> n; n is set to 5 None of these It does not compile. It compiles, but fails to link It throws a runtime exception It sets an error state in in.	

What prints when this code runs?
enum class Coin
{
PENNY = 1, NICKEL = 5, DIME = 10, QUARTER = 25
};
Coin c = NICKEL;
cout << static_cast<int>(c) << endl;
 	
5
Does not compile; Missing semicolon at end of list of members.
Does not compile; Cannot assign NICKEL to c.
2
	wrong
	Does not compile; Cannot assign NICKEL to c.
	Does not compile; Missing semicolon at end of list of members.
	
	i think its 5

What prints when this code runs?
enum class Coin
{
PENNY = 1, NICKEL, DIME, QUARTER
};
cout << static_cast<int>(Coin::DIME) << endl;
 	
10
Does not compile; Missing semicolon at end of list of members.
3
2
	...

Examine the following code (which is legal). Which statement is legal?
 	
struct Money { int dollars{0}, cents{0}; } m1, m2;
cout << m1 << endl;
 	
m1 = m2;
m2 = {3, 4}
if (m1 != m2) . . .
PreviousNext
	m1 = m2;

What is stored in data after this runs? vector<int> data{1, 2, 3}; data.back(); [1, 2] [2, 3] [1, 2, 3, 0] [1, 2, 3] [] None of these	[1, 2, 3]

{
vector<int> v{1, 2, 3};
for (auto i = v.size() - 1; i >= 0; i--)
cout << v.at(i) << " ";
cout << endl;
}
 	
Compiler error (does not compile)
Endless loop (may crash, but not necessarily)
Prints 3 2 1
Crashes when run
Issues a compiler warning, but no error
	Issues a compiler warning, but no error

Points to the first element in v
v.begin()
 	
Creates the empty vector []
vector<int> v;
 	
Safely returns a reference to the fourth element in v
v.at(3);
 	
Removes the last element in v
v.pop_back()
	v.begin()
	
	vector<int> v;
	
	v.at(3);
	
	v.pop_back()

Used to access the data inside a variable -> variable name Determines the amount of memory required and the operations permitted on a variable -> variable type The meaning assigned to a set of bits stored at a memory location -> variable value An object whose value is an address in memory -> pointer Expression using the address operator -> p = &a; Expression using the reference declarator -> int x = 3; Expression using the dereferencing operator -> y = *a; Expression using the pointer declarator -> double * v; Expression returning the number of allocated bytes used by an object -> sizeof(Star) Address value 0 -> nullptr	Used to access the data inside a variable -> variable name Determines the amount of memory required and the operations permitted on a variable -> variable type The meaning assigned to a set of bits stored at a memory location -> variable value An object whose value is an address in memory -> pointer Expression using the address operator -> p = &a; Expression using the reference declarator -> int x = 3; Expression using the dereferencing operator -> y = *a; Expression using the pointer declarator -> double * v; Expression returning the number of allocated bytes used by an object -> sizeof(Star) Address value 0 -> nullptr

cout << n << endl; The address value 0 No compilation errors, but undefined behavior No output; compiler error. The word "nullptr"	

Examine the following code. What is stored in a after it runs.
int f(int * p, int x)
{
p = x 2;
return x / 2;
}
. . .
int a = 3, b, c;
c = f(&b, a);
 	
2
3
Does not compile
1
6
	3

What prints?
int a[] = {1, 3, 5, 7, 9};
int *p = a;
cout << *++p;
cout << *p << endl;
 	
33
13
None of these
22
12
	33

What is printed when you run this code?
int n{};
int *p = &n;
*p = 10;
n = 20;
cout << *p << endl;
 	
10
20
The address of n
0
None of these
	20

Assume that ppi correctly points to pi. Which line prints the value stored inside pi?
int main()
{
double pi = 3.14159;
double *ppi;
// code goes here
// code goes here
}
 	
cout << *ppi;
None of these
cout << &ppi;
cout << ppi;
	cout << *ppi;

one
three
zero
two

Which fragment completes this code segment?

```
string fmt(double n, int decimals)
{
ostringstream out;
out << fixed << setprecision(decimals);
out << n;
return _________;
}
```

out.to_string()
n
None of these
out.str()
out

out.str()

What prints?

```
string s("hello");
try {
if (s.size() > 20) throw 42;
if (isupper(s.back())) throw "goodbye";
if (s == "Hello") throw string("hello");
s[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"; }
```

one
four
Undefined
five
two
three

one

Which line compiles, but crashes (throws an exception) when run?

```
int main()
{
vector<int> v{1, 2, 3};
auto size = v.size();

cout << v.back() << endl;// 1.
cout << v.front() << endl; // 2.
cout << v.at(0) << endl; // 3.
cout << v.at(size) << endl; // 4.
cout << v.pop_back() << endl; // 5
.}
```

4
5
1
2
3

4

cout << speed[5];
speed.front() = 12;
None of these
speed.erase(speed.begin());
speed[0] = speed.back()	

Which line has undefined output?
double speed[5] = {. . .};
 	
cout << speed[1] << endl;
cout << speed[4] << endl;
None of these
cout << speed[5] << endl;
cout << speed[0] << endl;
	cout << speed[5] << endl;

What is printed when you run this code? int *n{nullptr}; cout << n << endl; The word "nullptr" No output; compiler error. No compilation errors, but undefined behavior The address value 0	The address value 0
What is printed when you run this code? int *n{nullptr}; cout << *n << endl; No compilation errors, but undefined behavior The word "nullptr" The address value 0 No output; compiler error.	No compilation errors, but undefined behavior
What is x? vector<int> v{1, 2, 3}; auto x = max_element(v.begin(), v.end()); a std::pair object 3 an iterator 1 None of these	an iterator
What prints? vector<int> v{1, 2, 3, 4, 5}; cout << v.pop_back() << endl; 4 Nothing; compile-time error. 1 Nothing; run-time error. 5	Nothing; compile-time error.

```
vector<int> v{1, 2, 3};
auto size = v.size();
cout << v.back() << endl; // 1.
cout << v.front() << endl; // 2.
cout << v.at(0) << endl; // 3.
cout << v.at(size) << endl; // 4.
cout << v.pop_back() << endl; // 5.
}
```

1
5
4
3
2

Which area of memory are local variables stored in?

Heap
Initialized Data
Uninitialized Data
Stack
Text

Stack

Which line throws and out_of_range exception?double speed[5] = {. . .};

cout << speed[1] << endl;
cout << speed[0] << endl;
None of these
cout << speed[4] << endl;
cout << speed[5] << endl;

None of these

Which pointer initialization is illegal?

```
int a[ ] = {1, 3, 5, 7, 9};
```

int *p3 = &a[1];
int *p4 = &a;
int *p2 = a + 3;
int *p1 = a;
None of these

int *p4 = &a;

What prints?

```
string s("hello");
try {
if (s.size() > 2) throw 42;
if (isupper(s.back())) throw "goodbye";
if (s == "Hello") throw string("hello");
s[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"; }
```

one
four
Undefined
five
two

two

return
throw
None of these
catch
try
raise

Which of the following loop patterns are used here?

```
auto len = str.size();
while (len) out << str.at(--len);
```

counter-controlled loop
iterator or range loop
primed loop
inline test
limit loop
loop-and-a-half
data loop
sentinel loop

counter-controlled loop

What happens when you execute the (erroneous) line:
cout << stoi("fifteen") << endl;

Nothing happens. The statement is skipped

An exception is thrown, which may be caught.
If it is not caught, the program terminates.

The program prints an error message and continues running

The conversion is impossible, so the code will not compile.

The program prints an error message and unconditional terminates

An exception is thrown, which may be caught. If it is not caught, the program terminates.

What happens when you execute the following (erroneous) code:
cout << stoi(42.5) << endl;

The double 42.5 is truncated to 42 and printed

The code does not compile because the argument is the wrong type.

An exception is thrown, which may be caught

The program prints an error message and terminates since you cannot convert a double to an int

No conversion takes place and the output stream is placed in a failed state.

...

What is x?
vector<int> v{1, 2, 3};
auto x = *max_element(v.begin(), v.end());

1
None of these
a std::pair object

3

```
struct Rectangle { int length, width; } big, small;
```

None of these are correct
if (big == small) . . .
if (big.length == width) . . .
if (big != small) . . .
if (big.length == small.width) . . .

[1413] What does this loop do?

```
int a[ ] = {6, 1, 9, 5, 1, 2, 3};
int x(0);
for (auto e : a) x += e;
cout << x << endl;
```

Selects the smallest value in a
Counts the elements in a
Selects the largest value in a
Has no effect
Sums the elements in a

Sums the elements in a

What is true about an uninitialized pointer?

It is set to the nullptr value
None of these are true
Dereferencing it will cause a program crash
Dereferencing it is safe, but has no effect.
Dereferencing it is undefined behavior

Dereferencing it is undefined behavior

The value for the variable c is stored:

```
int a = 1;
void f(int b)
{
int c = 3;
static int d = 4;
}
```

in the CPU machine registers
on the heap
in the static storage area
on the stack
The example does not provide enough information

on the stack

The value for the variable d is stored:

```
int a = 1;
void f(int b)
{
int c = 3;
static int d = 4;
}
```

in the CPU machine registers
on the heap
in the static storage area
on the stack
The example does not provide enough information

in the static storage area

void swap(int * x, int & y)
{
. . .
}
. . .
int a = 3, b = 7;
 	
None of these
swap(a, &b);
swap(a, b);
swap(&a, b);
swap(&a, &b);	

All of these are legal C++ statements; which of them uses indirection?
 	
int a = 3, b = 4;
 	
None of these use indirection.
z *= a;
int y = a * b;
int x = *p;
int *p = &b;
	int x = *p;

What is the correct prototype for the output operator?
enum class Suit
{
HEARTS, SPADES, CLUBS, DIAMONDS
};
 	
ostream& operator>>(ostream& out, Suit& suit);
ostream& operator<<(ostream& out, Suit& suit);
ostream& operator<<(ostream& out, Suit suit);
ostream& operator<<(ostream& out, const Suit& suit);
	ostream& operator<<(ostream& out, Suit suit);

What is correct for # 3? (3 e) { cout << e.4 () << endl; catch if what exception& None of these try while	exception&
What happens when this code fragment runs? istringstream in("12"); int n; in >> n; It compiles, but fails to link It throws a runtime exception n is set to 12 None of these It sets an error state in in. It does not compile. PreviousNext	n is set to 12

____
inside a _______ block?

None of these
catch, try
raise, except
throw, catch
try, catch
if, else

When using the STL function count, the third argument is:

the value to count
cend(v)
cbegin(v)
None of these
a predicate function

...

Given the following structure and variable definitions, which data members are default initialized?

```
struct Employee
{
long empID;
std::string lastName;
double salary;
int age;
};
Employee bob;
```

None of these
salary
empID
lastName
age

age
empID
salary

[1426] What prints?

```
int a[] = {1, 3, 5, 7, 9};
int *p = a;
cout << ++*p;
cout << *p << endl;
```

13
12
None of these
22
33

22

Examine the following code (which is legal). What is the correct prototype for an aggregate output operator?

```
struct Time { int hours{0}, minutes{0}, seconds{0}; };
```

ostream& operator<<(ostream& out, const Time& m);

The value for the variable a is stored:

```
int a = 1;
void f(int b)
{
int c = 3;
static int d = 4;
}
```

in the static storage area

```
int x(100);
cout << &x << endl;
```

None of these

The value stored at address 100

The memory location where x is stored

The value of x (100)

After writing data to an ostringstream object named os, you can retrieve the string it contains by using:

None of these

to_string(os)

os.to_string()

os.str()

static_cast<string>(os);

Next

os.str()

What prints?

```
string s("hello");
try {
if (s.size() > 2) throw 42;
if (islower(s.back())) throw s.back();
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"; }
```

one

four

Undefined

five

two

three

five

What happens when this code fragment runs?cout << stoi("12") << endl;

It throws a runtime exception

It does not compile.

None of these

stoi() returns 12

It compiles, but fails to link

It sets an error state in cout.

stoi() returns 12

Assume s1 and s2 are C++ string objects. Which of these calls is illegal?

```
template <typename T>
void addem(T a, U b)
{
cout << a << " + " << b << "->"
<< (a + b) << endl;
}

addem(s1, s2);
```

None of these

addem(3, 4)

None of these

template <typename T>
void addem(T a, T b)
{
cout << a << " + " << b << "->"
<< (a + b) << endl;
}
 	
addem(s1, s2);
None of these
addem(3, 4)
addem(1.5, 2);
addem(4.5, 5.5);	

When using the STL function count_if, the third argument is: None of these the value to count cend(v) a predicate function cbegin(v)	None of these

Examine the following definition. empID is a ________.
 	
struct Employee
{
long empID;
std::string lastName;
double salary;
int age;
};
 	
instance variable
None of these
data member
field
structure tag
type-id
	data member

Given the following structure and variable definitions which statements are legal?
 	
struct Money
{
int dollars{0};
int cents{1};
};
Money payment;
 	
payment{1} = 5;
Money{1} = Money{0};
cout << Money.dollars;
payment.cents = 5;
cout << payment.dollars;
None of them
	payment.cents = 5;
	cout << payment.dollars;

Which area of memory are global variables stored in? Static storage area Heap Stack Text	Static storage area

```
char ch;
while (in.get(ch))
{
if (ch >= 'A' && ch <= 'Z')
upper++;
}
```

Group of answer choices
sentinel loop
counter-controlled loop
iterator or range loop
primed loop
inline test
limit loop
data loop
loop-and-a-half

A(n) __ is an occurrence of an undesirable situation that can be detected during program execution.

bug
exception
misfire
crash
PreviousNext

exception

What happens with the following section of code?

```
cout << "Enter 1, 2 or 3: ";
int n;
cin >> n;
#if 1
cout << "You entered 1" << endl;
#elif 2
cout << "You entered 2" << endl;
#elif 3
cout << "You entered 3" << endl;
#else
cout << "Invalid value" << endl;
#endif
```

Compiles, but only prints "Invalid value"
Does not compile
Compiles, but always print "You entered 1"
Compiles and prints the correct value entered by the user.
PreviousNext

Compiles, but always print "You entered 1"

Which of the following blocks is designed to catch any type of exception?

catch(...){ }

Which expression returns the number of countries?

```
string countries[] = {"Andorra", "Albania", . . . };
```

sizeof(countries) * sizeof(countries[0])
countries.length
sizeof(countries)
None of these
len(countries)

None of these

string countries[] = {"Andorra", "Albania", . . . };
 	
sizeof(countries)
len(countries)
sizeof(countries) / sizeof(string)
None of these
sizeof(countries) *
sizeof(countries[0])	

Which expression returns the number of countries?
 	
string countries[] = {"Andorra", "Albania", . . . };
 	
len(countries)sizeof(countries) * sizeof(countries[0])
sizeof(countries)
None of these
sizeof(countries) / sizeof(countries[0])
	sizeof(countries) / sizeof(countries[0])

Assume that p is a pointer to the first of 50 contiguous integers stored in memory. What is the address of the first integer appearing after this sequence of integers?
 	
p + sizeof(int) * 50;
None of these
&p + 50;
p + 50;
sizeof(p) + 50;
	p + 50;

Which array definition produces {0, 1, 2}?
 	
int SIZE = 3;
int a1[SIZE];
int a2[3];
int a3[3]{};
int a4[] = {1, 2, 3};
int a5[3] = {1, 2};
 	
a1
None of these
a2
a5
a3
	None of these

Which returns the last pixel on the first row of this image?
Pixel *p; // address of
pixel dataint w, h; // width and height of image
 	
None of these are correct
p + w - 1
p[w - 1]
p[w] - 1
*p[w - 1]
	p[w - 1]

int SIZE = 3;
int a1[SIZE];
int a2[3];
int a3[3]{};
int a4[] = {1, 2, 3};
int a5[3] = {1, 2};
 	
a3
a1
a5
a2
None of these
PreviousNext	

Assume that p1 is a pointer to an integer and p2 is a pointer to a second integer. Both integers appear inside a large contiguous sequence in memory, with p2 storing a larger address. How many total integers are there in the slice between p1 and p2?
 	
p1 - p2 + 1;
None of these
p2 - p1;
p2 - p1 - 1;
p1 - p2;
	p2 - p1;

The try block is followed by one or more ___ blocks. throw catch do finally	catch
A(n) __________ is a statement about a condition which must be true when it is encountered in your code. precondition assertion postcondition exception PreviousNext	assertion
Read and write characters to memory using streams ->sstream Connect a disk file to an input or output stream. ->fstream Use the predefined stream objects cin and cout ->iostream Determine the category of a character ->cctype Modify the way that memory is converted to characters on input or output ->iomanip	Read and write characters to memory using streams ->sstream Connect a disk file to an input or output stream. ->fstream Use the predefined stream objects cin and cout ->iostream Determine the category of a character ->cctype Modify the way that memory is converted to characters on input or output ->iomanip

```
struct Rectangle { int length, width; } big, little;

cin >> big;
double p = 2 * (length + width);
None of these are correct
cin >> little.width;
cout << Rectangle.length;
```

Assume vector<double> speed(5); Which line throws a runtime error?

```
speed.front() = 12;
speed[0] = speed.back()
None of these
cout << speed[speed.size()];
speed.erase(speed.begin());
```

speed.erase(speed.begin());

What is printed when you run this code?

```
int *n{nullptr};
cout << &n << endl;
```

The address value 0
No compilation errors, but undefined behavior
No output; compiler error.
The address value where n is stored
The word "nullptr"

The address value where n is stored

The variable buf is a pointer to a region of memory storing contiguous int values. (This is similar to your homework, where you had a region of memory storing unsigned char values.) The four lines shown here are legal. Which operation is legal?

```
int *p1 = buf;
const int *p2 = buf;
int * const p3 = buf;
const int * p4 const = buf;

*p2 = 3;
p4++;
*p4 = 7;
p3++;
*p3 = 5;
```

*p3 = 5;

Which line creates an array with 5 elements?

int b[5];

Which of these lines correctly prints 3?

```
struct S {
int a = 3;
double b = 2.5;
};

S obj, *p = &obj;

cout << p.a << endl;
cout << (*p).a << endl;
cout << *p.a << endl;
cout << *(p.a) << endl;
cout << *(p).a << endl;
```

cout << (*p).a << endl;

```
vector<int> *vp(NULL);
double *pd = 0;
int *pi = nullptr;
Star *ps = NULL;
```
All are equally preferred.

What is stored in the last element of nums?

```
int nums[3] = {1, 2};
```

Syntax error in array declaration
2
0
Undefined value
1

0

What happens when you execute the (erroneous) line:
```
bool ok = 2 + 2 == 5;
```

A)The program prints an error message and the program continues
B)An exception is thrown because 2 + 2 does not equal 5
C)The code does not compile, since 2 + 2 is not equal to 5
D)The program prints an error message and terminates
E)Nothing happens. The variable ok is set to false.

...

What happens when you execute the (erroneous) line:
```
auto x = sqrt(-1);
```

The code does not compile. You cannot take the square root of a negative number.
The statement is skipped and the program continues
The function returns an error value and the program continues
The program prints an error message and terminates
The function throws and exception. If not caught the program terminates.

The function returns an error value and the program continues

---

What prints?
```
string s("hello");
try {
if (s.size() > 5) throw s.size();
if (isupper(s.back())) throw s.back();
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"; }
```

one
four
Undefined

two

```
char ch;
in.get(ch);
while (ch != 'Q')
{
pos++;
in.get(ch);
}
```

sentinel loop
inline test
limit loop
data loop
primed loop
iterator or range loop
counter-controlled loop
loop-and-a-half