

Midterm 3 Study Guide

Due No due date Points 25 Questions 25 Time Limit 30 Minutes Allowed Attempts Unlimited

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 30	23 minutes	25 out of 25
	Attempt 30	23 minutes	25 out of 25
LATEST	Attempt 29	23 minutes	24 out of 25
	Attempt 28	22 minutes	25 out of 25
	Attempt 27	20 minutes	20 out of 25
	Attempt 26	16 minutes	25 out of 25
	Attempt 25	18 minutes	24 out of 25
	Attempt 24	15 minutes	25 out of 25
	Attempt 23	21 minutes	24 out of 25
	Attempt 22	24 minutes	24 out of 25
	Attempt 21	18 minutes	24 out of 25
	Attempt 20	30 minutes	19 out of 25
	Attempt 19	20 minutes	23 out of 25
	Attempt 18	30 minutes	21 out of 25
	Attempt 17	30 minutes	23.5 out of 25
	Attempt 16	30 minutes	24 out of 25
	Attempt 15	30 minutes	21 out of 25
	Attempt 14	30 minutes	23 out of 25
	Attempt 13	24 minutes	22 out of 25
	Attempt 12	16 minutes	24 out of 25
	Attempt 11	17 minutes	19 out of 25
	Attempt 10	17 minutes	22 out of 25
	Attempt 9	20 minutes	20 out of 25
	Attempt 8	21 minutes	20 out of 25
	Attempt 7	25 minutes	21.5 out of 25
	Attempt 6	25 minutes	21 out of 25
	Attempt 5	30 minutes	17 out of 25
	Attempt 4	21 minutes	23 out of 25
	Attempt 3	26 minutes	19.89 out of 25
	Attempt 2	30 minutes	22.5 out of 25
	Attempt 1	27 minutes	21 out of 25



ⓘ Correct answers are hidden.

Submitted Jul 20 at 4:55pm

Question 11 / 1 pts

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.

- ☐ True
- ☒ False

Question 21 / 1 pts

The order of the `catch` blocks does not affect the program.

- ☐ True
- ☒ False

Question 31 / 1 pts

What is true about this piece of code?

```
template <typename T, typename U>
T pickle(T& a, const U& b) {
    a += b;
    return b;
}

int main()
{
    int x = 42;
    auto a = pickle(x, 4.5);
    cout << a << endl;
    cout << x << endl;
}
```

- ☒ In main, a prints 4
- ☐ This code has a syntax error.
- ☐ In main, a prints 4.5
- ☒ In main, x prints 46
- ☐ In main, x prints 46.5



Question 4

1 / 1 pts

The statement `#if abs(-3) > 2` is legal.

- ☐ True
- ☒ False

Question 5

1 / 1 pts

When you throw an exception, control immediately jumps out of the current try block.

- ☒ True
- ☐ False

Question 6

1 / 1 pts

The directives `#if defined(symbol)` and `#ifndef symbol` mean, essentially, the same thing.

- ☐ True
- ☒ False

Question 7

1 / 1 pts

Which call below produces 5?

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

- ☐ None of these
- ☐ `addem(3.0, 2.5)`
- ☒ `addem<int>(3, 2.5);`
- ☐ `addem(3, 2.5);`
- ☐ `addem<double>(3, 2.5);`

Question 8

1 / 1 pts

What is correct for # 4?

```
int main()
{
    1 {
        string s = "hello";
        cout << s.at(5) << endl;
    }
    2 ( 3 e)
    {
        cout << e. 4 () << endl;
    }
}
```

- ☐ try
- ☐ exception&
- ☐ None of these
- ☒ what
- ☐ while
- ☐ catch
- ☐ if



Question 9

1 / 1 pts

To use different versions of a function depending on the platform is called *conditional compilation*.

- ☒ True
- ☐ False

Question 10

1 / 1 pts

The C++ specific term for classes like vector are template classes.

- ☒ True
- ☐ False

Question 11

1 / 1 pts

vector subscripts begin at 0 and go up to the vector size.

- ☐ True
- ☒ False

Question 12

1 / 1 pts

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

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

Question 13

1 / 1 pts

Assume the vector `v` contains `[1, 2, 3]`. `v.pop_back();` changes `v` to `[1, 2]`.

- ☒ True
- ☐ False

Question 14

1 / 1 pts

Assume the vector `v` contains `[1, 2, 3]`. `v.erase(0);` is a syntax error.

- ☒ True

☐ False

Question 151 / 1 pts

The STL or Standard Template Library is a collection of data structures and algorithms developed by?

☒ Alexander Stepanov

☐ Dennis Ritchie

☐ Bjarne Stroustup

☐ John Von Neumann

☐ Ken Thompson

Question 161 / 1 pts

Assume the vector v contains [1, 2, 3]. v.erase(0); changes v to [2, 3].

☐ True

☒ False



Question 171 / 1 pts

The declaration: vector<string> v(5); creates a vector containing five empty string objects.

☒ True

☐ False

Question 181 / 1 pts

If *p* is a pointer to a structure, and the structure contains a data member *x*, you can access the data member by using the notation: **p->x*

☐ True

☒ False

Question 191 / 1 pts

Explicitly initializing an array like this: *int a[3] = {1, 2, 3};* requires the size and the number of elements supplied to be the same.

☐ True

☒ False

Question 201 / 1 pts

What is printed when you run this code?

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

☐ The word "nullptr"

☐ No compilation errors, but undefined behavior

☐ The address value 0

☒ The address value where *n* is stored

☐ No output; compiler error.

Question 211 / 1 pts

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 illegal?*

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

☐ *p1 = 3;

☐ *p3 = 5;

☐ p1++;

☐ p2++;

☒ *p2 = 7;

Question 22

1 / 1 pts

What is true about this code?

```
int * choice;
```

☐ Syntax error; should be int choice*;

☐ choice currently points to an integer

☒ choice contains an undefined address

☐ choice currently contains an integer

☐ choice can point to any kind of object

Question 23

1 / 1 pts

These pointers should point to "nothing". Which is not correctly initialized?

☐ int *pi = nullptr;

☐ Star *ps = NULL;

☐ All are correctly initialized to point to nothing.

☐ double *pd = 0;;

☒ vector<int> *vp;

Question 24

1 / 1 pts

What is the term used to describe a variable which stores a memory address?

☐ lvalue

☒ pointer

☐ None of these

☐ rvalue

☐ reference

Question 25

1 / 1 pts

Which line creates an array with 5 elements?

☐ None of these

☐ int[] c[5];

☐ int a[4];

☐ int[5] d;

☒ int b[5];

