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 12	16 minutes	24 out of 25
LATEST	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 13 at 6:32am

Question 1

1 / 1 pts

Which of the following loop patterns are used here?

```
size_t pos = 0;
char ch;
in.get(ch);
while (ch != 'Q')
{
    pos++;
    in.get(ch);
}
```

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

Question 2

1 / 1 pts

The function ____ returns a string containing an appropriate message.

- `where`
- `what`
- `log`
- `when`

Incorrect

Question 3

0 / 1 pts

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);`
- `addem(3, 4)`

- (●) `addem(1.5, 2);`
- ( ) `addem(4.5, 5.5);`
- ( ) None of these

Question 4

1 / 1 pts

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

- ( ) True
- (●) False

Question 5

1 / 1 pts

The line: `ifstream in("x");` throws a runtime exception if a file `x` cannot be found.

- ( ) True
- (●) False

Question 6

1 / 1 pts

Which fragment completes this code segment?

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

- ( ) None of these
- ( ) `out`
- (●) `out.str()`
- ( ) `n`
- ( ) `out.to_string()`

Question 7

1 / 1 pts

Which of the following loop patterns are used here?

```
string s{"hello CS 150"};
for (auto e : s)
{
    if (toupper(e)) break;
}
```

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

Question 8

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<double>(3, 2.5);`
- `addem<int>(3, 2.5);`
- `addem(3, 2.5);`
- `addem(3.0, 2.5)`

Question 9

1 / 1 pts

Functions with generic parameters may use the keyword `class` or the keyword `typename` for their type parameters.

- True
- False

Question 10

1 / 1 pts

The statement `v.insert(v.begin(), 3)` inserts the element `3` into the `vector` `v`, overwriting the exiting element at index `0`.

- True
- False

Question 11

1 / 1 pts

What is x?

```
vector<int> v{1, 2, 3};
auto x = min_max_element(v.begin(), v.end());
```

- None of these
- `1`
- a `std::pair` object
- an iterator
- `3`

Question 12

1 / 1 pts

A `vector` subscript represents the element's counting position.

- True
- False

Incorrect

Question 13

0 / 1 pts

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

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

Employee bob{};
```

- [ ] `empID`
- [x] None of these
- [ ] `salary`
- [ ] `lastName`
- [ ] `age`

Question 14

1 / 1 pts

The following code is ***legal***.

```
struct {int hours, seconds; } MIDNIGHT{0, 0};
```

- (●) True
- ( ) False

Question 15

1 / 1 pts

The declaration: `vector<int> v(10, 5);` creates a `vector` object containing five integers.

- ( ) True
- (●) False

Question 16

1 / 1 pts

What prints?

```
void f(vector<int> v)
{
    v.at(0) = 42;
}
int main()
{
    vector<int> x{1, 2, 3};
    f(x);
    cout << x.at(0) << endl;
}
```

- ( ) Nothing; run-time error.
- ( ) Nothing; linker error
- ( ) Nothing; compile-time error.
- (●) 1
- ( ) 42

Question 17

1 / 1 pts

What is stored in `data` after this runs?

```
vector<int> data{1, 2, 3};
data.back();
```

- ( ) [1, 2, 3, 0]
- (●) [1, 2, 3]
- ( ) [1, 2]
- ( ) None of these
- ( ) [2, 3]
- ( ) []

Question 18

1 / 1 pts

The size of the array is stored along with its elements.

- ( ) True
- (●) False

Question 19

1 / 1 pts

In C++ using == to compare one array to another is permitted (if meaningless).

- (●) True
- ( ) False

Question 20

1 / 1 pts

The elements of a C++ array created in a function are allocated on the stack.

- [x] True
- [ ] False

Question 21

1 / 1 pts

A forward reference can be used when you want to use a structure as a data member without first defining the entire structure.

- [ ] True
- [x] False

Question 22

1 / 1 pts

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

- [x] `vector<int> *vp;`
- [ ] `Star *ps = NULL;`
- [ ] `int *pi = nullptr;`
- [ ] `double *pd = 0;;`
- [ ] All are correctly initialized to point to nothing.

Question 23

1 / 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
- [x] False

Question 24

1 / 1 pts

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

- [ ] `vector<int> *vp(0);`
- [ ] `int *pi = nullptr;`
- [ ] `Star *ps = NULL;`
- [ ] `double *pd{};`
- [x] All are correctly initialized to point to nothing.

Incorrect

Question 25

0 / 1 pts

What is printed when you run this code?

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

- [ ] No output; compiler error.
- [ ] The address value `0`
- [x] No compilation errors, but undefined behavior
- [ ] The word "`nullptr`"