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 28	22 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 2:45pm

Question 1

1 / 1 pts

Which of the following loop patterns are used here?

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

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

Question 2

1 / 1 pts

A loop that reads data until the input stream signals that it is done is called a data loop.

- True
- False

Question 3

1 / 1 pts

In the ***flag-controlled-pattern***, you read data before the loop and at the end of the loop.

- True
- False (selected)

Question 4

1 / 1 pts

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

- `catch(){ }`
- `catch(exception){ }`
- `catch(...){ }` (selected)
- `catch(*){ }`

Incorrect

Question 5

0 / 1 pts

Function templates with generic parameters may use the keyword `class` or the keyword `struct` for their type parameters.

- True (selected)
- False

Question 6

1 / 1 pts

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

- crash
- exception (selected)
- bug
- misfire

Question 7

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;
    }
}
```

- `exception&`
- `try`
- `catch`
- `what` (selected)
- `if`
- None of these
- `while`

Question 8

1 / 1 pts

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

- True (selected)
- False

Question 9

1 / 1 pts

What is true about this code?

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

int main()
{
    auto x = 42.0;
    auto y = pickle(x, 4.5);
    cout << x << endl;
    cout << y << endl;
}
```

- [ ] This code does not compile.
- [ ] In main, x prints 46
- [ ] In main, y prints 4
- [x] In main, x prints 46.5
- [x] In main, y prints 4.5

Question 10 1 / 1 pts

The following is an ***anonymous structure***.

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

- (•) True
- ( ) False

Question 11 1 / 1 pts

An unnamed (anonymous) function is called a(n):

- ( ) functor
- ( ) iterator
- (•) lambda
- ( ) stub
- ( ) None of these

Question 12 1 / 1 pts

A `vector` represents a linear heterogeneous collection of data.

- ( ) True
- (•) False

Question 13 1 / 1 pts

Examine the following code (which is legal). Which statement is ***legal***?

```cpp
struct Money { int dollars{0}, cents{0}; } m1, m2;
```

- ( ) `m2 = {3, 4}`
- ( ) `if (m1 != m2) . . .`
- ( ) `cout << m1 << endl;`
- (•) `m1 = m2;`

Question 14 1 / 1 pts

The statement `v.insert(v.begin(), 3)` inserts the element `3` into the `vector` `v`, shifting the existing elements to the right.

- (•) True

False

Question 15 1 / 1 pts

The built-in primitive data types such as `int`, char and `double` are ***structured*** data types.

True

False

Question 16 1 / 1 pts

The declaration: `vector<int> v = new vector<>();` creates a `vector` object with no elements.

True

False

Question 17 1 / 1 pts

Match each item with the correct statement below.

Creates the vector [0]	vector<int> v(1);
Returns a reference to the fourth element in v with no range checking.	v[3];
Creates the vector [3, 3]	vector<int> v(2, 3);
Adds a new element to the end of v	v.push_back(3);

Question 18 1 / 1 pts

The value for the variable ***d*** is stored:

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

on the heap

on the stack

in the CPU machine registers

The example does not provide enough information

in the static storage area

Question 19 1 / 1 pts

An incomplete type and a forward reference generally mean the same thing.

True

False

Question 20 1 / 1 pts

The value for the variable ***a*** is stored:

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

in the CPU machine registers

on the stack

- on the heap
- in the static storage area
- The example does not provide enough information

Question 21

1 / 1 pts

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 + 50;`
- None of these
- `sizeof(p) + 50;`
- `p + sizeof(int) * 50;`
- `p + 50;`

Question 22

1 / 1 pts

Examine this version of the ***swap()*** function. How do you call it?

```
void swap(int * x, int & y)
{
    . . .
}
. . .
int a = 3, b = 7;
// What goes here ?
```

- `swap(a, b);`
- `swap(a, &b);`
- None of these
- `swap(&a, b);`
- `swap(&a, &b);`

Question 23

1 / 1 pts

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

- True
- False

Question 24

1 / 1 pts

The value for the variable ***c*** is stored:

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

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

Question 25

1 / 1 pts

Match each item with the correct term below.

Expression using the address operator	p = &a;
Expression using the dereferencing operator	y = *a;

Expression returning the number of allocated bytes used by an object	sizeof(Star)
Address value 0	nullptr