

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 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 2:21pm

Question 1

1 / 1 pts

Match each item with the correct loop form below.

Indefinite limit loop that reduces its input	<code>while (n != 0) { n /= 2; }</code>
Indefinite limit loop that uses successive approximations	<code>while(abs(g1 – g2) >= EPSILO)</code>
Counter-controlled symmetric loop for producing a sequence of data	<code>for (int i=12; i <= 19; i++) { . . }</code>
Indefinite data loop that uses raw input	<code>while(cin.get(ch)) { . . }</code>
Counter-controlled asymmetric loop for processing characters	<code>for (size_t i=0, len=s.size(); i < l</code>
Iterator loop that may change its container	<code>for (auto& e : col) { . . }</code>
Iterator loop that cannot change its container	<code>for (auto e : col) { . . }</code>
Counter-controlled loop for processing substrings	<code>for (size_t i=4, slen=4, len=s.siz</code>
Indefinite data loop that uses formatted input	<code>while(cin >> n) { . . }</code>

Question 2

1 / 1 pts

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

- ☐ True
- ☒ False



Question 31 / 1 pts

What happens when you execute the (erroneous) line:

assert(2 + 2 == 5);

☐ Nothing happens. The statement is not true, so it is skipped

☒ The program prints an error message and terminates

☐ Since this is impossible, the code will not compile.

☐ An exception is thrown, which may be caught.

☐ The program prints an error message and continues running

Question 41 / 1 pts

A loop that reads data until some special value is found is called a sentinel loop.

☒ True

☐ False

Question 51 / 1 pts

You compiler or contains constants that can be used to identify the platform you are compiling on.

☒ True

☐ False

Question 61 / 1 pts

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

☐ True

☒ False

Question 71 / 1 pts

The `try` block is followed by one or more ____ blocks.

☐ `finally`

☐ `throw`

☒ `catch`

☐ `do`

Question 81 / 1 pts

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 string& e) { cout << "two\n"; }
catch (exception& e) { cout << "three\n"; }
catch (...) { cout << "four\n"; }
```

☐ three

☐ four

☒ two

☐ Undefined

☐ one



Question 91 / 1 pts

The function ____ returns a string containing an appropriate message.

☐ log

☒ what

☐ where

☐ when

Question 101 / 1 pts

In the declaration: vector<int> v; the word vector represents the object's **base type**.

☐ True

☒ False

Question 111 / 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;
}
```

☐ 1

☐ Nothing; run-time error.

☐ Nothing; linker error

☐ Nothing; compile-time error.

☒ 42

Question 121 / 1 pts

You may create a structure variable as part of a structure definition.

☒ True

☐ False

Question 131 / 1 pts

What prints?

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

☒ Nothing; compile-time error.

☐ Nothing; linker error

☐ 1

☐ 42

☐ Nothing; run-time error.

Question 141 / 1 pts

Examine the following code (which **is legal**). What changes are necessary to allow the statement `if (m1 != m2) ...` to compile?

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

bool equals(const Money& lhs, const Money& rhs)
{
    return lhs.cents == rhs.cents &&
           lhs.dollars == rhs.dollars;
}
```

- ☐ This is not possible in C++.
- ☐ The function `equals()` must be named `notEquals()`.
- ☐ The type `Money` needs to be a `class`
- ☐ The name of `equals()` must be changed to `operator==`
- ☒ You must write a function named `operator!=`

Question 151 / 1 pts

This is the correct syntax for a C++ scoped enumeration.

```
enum class WEEKEND {SUNDAY, SATURDAY=6};
```

- ☒ True
- ☐ False

Question 161 / 1 pts

What prints when this code runs?

```
enum class Coin
{
    PENNY = 1, NICKEL = 5, DIME = 10, QUARTER = 25
};

Coin c = Coin::NICKEL;
cout << static_cast<int>(c) << endl;
```

- ☐ 2
- ☐ Does not compile; Cannot assign `Coin::NICKEL` to `c`.
- ☒ 5
- ☐ Does not compile; Missing semicolon at end of list of members.

Question 171 / 1 pts

In C++, objects have **reference semantics**; object variables refer to, but do not contain the data members.

- ☐ True
- ☒ False

Question 181 / 1 pts

What is true about this code?

```
int n{500};
int *p = &n;
```

- ☐ `&n` is the indirect value of `p`
- ☒ `*p` is the value of `n`
- ☐ `p` stores the same value as `n`
- ☐ `&p` is the direct or explicit value of `n`
- ☐ `&p` represents the indirect value of `n`

Question 191 / 1 pts

The size of the array is **not** stored along with its elements.

- ☒ True



☐ False

Question 20

1 / 1 pts

In C++ assigning one array to another is illegal.

☒ True

☐ False

Question 21

1 / 1 pts

Match each item with the correct term below.

Determines the amount of memory required and the operations permitted on a variable

variable type



An object whose value is an address in memory

pointer



Expression using the reference declarator

int& x = 3;



Expression using the pointer declarator

double * v;



Question 22

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 + sizeof(int) * 50;

☒ p + 50;

☐ sizeof(p) + 50;

☐ None of these

☐ &p + 50;

Question 23

1 / 1 pts

All of these are legal C++ statements; which of them uses the C++ *dereferencing operator*?

`int a = 3, b = 4;`

☐ None of these use the dereferencing operator.

☐ z *= a;

☐ int y = a * b;

☐ int *p = &b;

☒ int x = *p;

Question 24

1 / 1 pts

The subscripts of a C++ array range from 0 to the allocated array size –1.

☒ True

☐ False

Question 25

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

☐ False