Chapter 13

Question 1 6.25 / 6.25 points

The constructor function may not accept arguments.

- ○ True
- ✓ ◉ False

Question 2 6.25 / 6.25 points

Members of the class object are accessed with the

- ✓ ◉ dot operator
- ○ `cin` object
- ○ extraction operator
- ○ stream insertion operator

- ○ None of these

Question 3 6.25 / 6.25 points

A class is a(n) _____ that is defined by the programmer.

- ✓ ◉ data type
- ○ function
- ○ method
- ○ attribute
- ○ None of these

Question 4 6.25 / 6.25 points

Objects are created from abstract data types that encapsulate _____ and _____ together.

- ○ numbers, characters
- ✓ ◉ data, functions
- ○ addresses, pointers
- ○ integers, floating-point numbers
- ○ None of these

Question 5 6.25 / 6.25 points

When a constructor function accepts no arguments, or does NOT have to accept arguments because of default arguments, it is called a(n)

- ○ empty constructor
- ✓ ◉ default constructor
- ○ stand-alone function
- ○ arbitrator function

- ○ None of these

Question 6 6.25 / 6.25 points

If you do NOT declare an access specification, the default for members of a class is

- ○ `inline`
- ✓ ◉ `private`
- ○ `public`
- ○ `global`
- ○ None of these

Question 7 6.25 / 6.25 points

Where are class declarations usually stored?

- ○ on separate disk volumes
- ✓ ◉ in their own header files
- ○ in .cpp files, along with function definitions
- ○ under pseudonyms
- ○ None of these

Question 8 6.25 / 6.25 points

A C++ class is similar to a(n)

- ○ inline function
- ○ header file
- ○ library function
- ✓ ◉ structure

- ○ None of these

Question 9 6.25 / 6.25 points

Examples of access specifiers are the key words

- ○ **`near`** and **`far`**
- ○ **`opened`** and **`closed`**
- ✓ ◉ **`private`** and **`public`**
- ○ **`table`** and **`row`**
- ○ None of these

Question 10 6.25 / 6.25 points

The destructor function's return type is

- ○ **`int`**
- ○ **`float`**
- ○ **`char`**
- ✓ ◉ Nothing; destructors have no return type
- ○ None of these

Question 11 6.25 / 6.25 points

In OOP terminology, an object's member variables are often called its _____ and its member functions can be referred to as its behaviors or its _____.

- ○ values, morals
- ○ data, activities
- ○ attributes, activities
- ✓ ◉ attributes, methods

- ○ None of these

Question 12 6.25 / 6.25 points

Class objects can be defined prior to the class declaration.

- ○ True
- ✓ ◉ False

Question 13 6.25 / 6.25 points

How many default constructors can a class have?

- ✓ ◉ only one
- ○ two or more
- ○ only two
- ○ any number
- ○ None of these

Question 14 6.25 / 6.25 points

Which of the following is a directive used to create an "include guard" that allows a program to be conditionally compiled to prevent a header file from accidentally being included more than once?

- ○ `#include`
- ○ `#guard`
- ✓ ◉ `#ifndef`
- ○ `#endif`
- ○ None of these

Question 15 6.25 / 6.25 points

The type of member function that may be called from a statement outside the class is

- ✔ ◉ **`public`**
- ○ **`private`**
- ○ undeclared
- ○ **`global`**
- ○ None of these

Question 16 6.25 / 6.25 points

In a procedural program you typically have _____ stored in a collection of variables and a set of _____ that perform operations on the data.

- ○ numbers, arguments
- ○ parameters, arguments
- ○ strings, operators
- ✔ ◉ data, functions
- ○ None of these

Done

Chapter 9

Question 1 4 / 4 points

If a variable uses more than one byte of memory, for pointer purposes its address is
If a variable uses more than one byte of memory, for pointer purposes its address is
Question options:

the address of the last byte of storage
the address of the last byte of storage

the average of all the addresses used to store that variable
the average of all the addresses used to store that variable

the address of the first byte of storage
the address of the first byte of storage

the address of the second byte of storage
the address of the second byte of storage

None of these
None of these

Question 2 4 / 4 points

C++ does not perform array bounds checking, making it possible for you to assign a pointer the address of an element out of the boundaries of an array.
C++ does not perform array bounds checking, making it possible for you to assign a pointer the address of an element out of the boundaries of an array.
Question options:

True

False

Question 3 4 / 4 points

A pointer can be used as a function argument, giving the function access to the original argument.

A pointer can be used as a function argument, giving the function access to the original argument.
Question options:

True

False

Question 4 4 / 4 points

After the code shown executes, which of the following statements is TRUE?

```
int numbers[] = {0, 1, 2, 3, 4};
int *ptr = numbers;
ptr++;
```

After the code shown executes, which of the following statements is TRUE?

```
int numbers[] = {0, 1, 2, 3, 4};
int *ptr = numbers;
ptr++;
```

Question options:

ptr will hold the address of **numbers[0]**

ptr will hold the address of **numbers[0]**

ptr will hold the address of the second byte within the element **numbers[0]**

ptr will hold the address of the second byte within the element **numbers[0]**

ptr will hold the address of **numbers[1]**

ptr will hold the address of **numbers[1]**

this code will not compile

this code will not compile

Question 5 4 / 4 points

Not all arithmetic operations can be performed on pointers. For example, you cannot _____ or _____ pointers.

Not all arithmetic operations can be performed on pointers. For example, you cannot _____ or _____ pointers.

Question options:

multiply, divide
multiply, divide

+=, -=
+=, -=

add, subtract
add, subtract

increment, decrement
increment, decrement

None of these
None of these

Question 6 4 / 4 points

A function may return a pointer but the programmer must ensure that the pointer

A function may return a pointer but the programmer must ensure that the pointer

Question options:

still points to a valid object after the function ends
still points to a valid object after the function ends

has not been assigned an address
has not been assigned an address

was received as a parameter by the function
was received as a parameter by the function

has not previously been returned by another function
has not previously been returned by another function

None of these
None of these

Question 7 4 / 4 points

A pointer variable is designed to store
A pointer variable is designed to store
Question options:

any legal C++ value
any legal C++ value

only floating-point values
only floating-point values

an integer
an integer

a memory address
a memory address

None of these
None of these

Question 8 4 / 4 points

Select all that apply. Of the following, which statements have the same meaning?
Select all that apply. Of the following, which statements have the same meaning?
Question options:

```
int *ptr = nullptr;
```

```
int *ptr = nullptr;
```


```
*int ptr = nullptr;
```

```
*int ptr = nullptr;
```


```
int ptr* = nullptr;
```

```
int ptr* = nullptr;
```


```
int* ptr = nullptr;
```

```
int* ptr = nullptr;
```


```
int ptr = nullptr;
```

```
int ptr = nullptr;
```


Question 9 4 / 4 points

A pointer variable may be initialized with
A pointer variable may be initialized with
Question options:

any nonzero integer value
any nonzero integer value

a valid address in the computer's memory
a valid address in the computer's memory

an address less than zero
an address less than zero

any nonzero number
any nonzero number

None of these
None of these

Question 10 4 / 4 points

Dynamic memory allocation occurs
Dynamic memory allocation occurs
Question options:

when a new variable is created by the compiler
when a new variable is created by the compiler

when a new variable is created at runtime
when a new variable is created at runtime

when a pointer fails to dereference the right variable
when a pointer fails to dereference the right variable

when a pointer is assigned an incorrect address
when a pointer is assigned an incorrect address

None of these
None of these

Question 11 4 / 4 points

If you are using an older computer that does NOT support the C++11 standard, you should initialize pointers with
If you are using an older computer that does NOT support the C++11 standard, you should initialize pointers with
Question options:

the integer **0** or the value **NULL**
the integer **0** or the value **NULL**

the null terminator **'\0'**
the null terminator **'\0'**

a nonzero value
a nonzero value

Any of these
Any of these

None of these
None of these

Question 12 4 / 4 points

Assuming **ptr** is a pointer variable, what will the following statement output?
cout << *ptr;
Assuming **ptr** is a pointer variable, what will the following statement output?
cout << *ptr;
Question options:

the value stored in the variable whose address is contained in **ptr**
the value stored in the variable whose address is contained in **ptr**

the string **"*ptr"**
the string **"*ptr"**

the address of the variable whose address is stored in **ptr**
the address of the variable whose address is stored in **ptr**

the address of the variable stored in **`ptr`**
the address of the variable stored in **`ptr`**

None of these
None of these

Question 13 0 / 4 points

In C++11, the **`nullptr`** keyword was introduced to represent the address **`0`**.
In C++11, the **`nullptr`** keyword was introduced to represent the address **`0`**.

Question options:

True

False

Question 14 4 / 4 points

The _____ and _____ operators can be used to increment or decrement a pointer variable.
The _____ and _____ operators can be used to increment or decrement a pointer variable.

Question options:

addition, subtraction
addition, subtraction

`++, --`
`++, --`

modulus, division
modulus, division

All of these
All of these

None of these
None of these

Question 15 4 / 4 points

Every byte in the computer's memory is assigned a unique
Every byte in the computer's memory is assigned a unique

Question options:

pointer
pointer

address
address

dynamic allocation
dynamic allocation

name
name

None of these
None of these

Question 16 4 / 4 points

The _____, also known as the address operator, returns the memory address of a variable.
The _____, also known as the address operator, returns the memory address of a variable.

Question options:

asterisk (*****)
asterisk (*****)

ampersand (**&**)
ampersand (**&**)

percent sign (**%**)
percent sign (**%**)

exclamation point (**!**)
exclamation point (**!**)

None of these
None of these

Question 17 4 / 4 points

Select all that apply. Select as many of the following options that make this sentence TRUE:
The contents of pointer variables may be changed with mathematical statements that perform

Select all that apply. Select as many of the following options that make this sentence TRUE:
The contents of pointer variables may be changed with mathematical statements that perform
Question options:

multiplication
multiplication

division
division

modulus
modulus

subtraction
subtraction

addition
addition

Question 18 Glitched 0 / 4 points

Assuming **`myValues`** is an array of **`int`** values and **`index`** is an **`int`** variable, both of the following statements do the same thing.

Assuming **`myValues`** is an array of **`int`** values and **`index`** is an **`int`** variable, both of the following statements do the same thing.

Question options:

True

False

Question 19 4 / 4 points

It is legal to subtract a pointer variable from another pointer variable.

It is legal to subtract a pointer variable from another pointer variable.
Question options:

True

False

Question 20 4 / 4 points

In the following statement, what does **`int`** mean?

```
int *ptr = nullptr;
```

In the following statement, what does **`int`** mean?

```
int *ptr = nullptr;
```

Question options:

The variable named **`*ptr`** will store an integer value.
The variable named **`*ptr`** will store an integer value.

The variable named **`*ptr`** will store an asterisk and an integer value
The variable named **`*ptr`** will store an asterisk and an integer value

`ptr` is a pointer variable and will store the address of an integer variable.
`ptr` is a pointer variable and will store the address of an integer variable.

The variable named **`*ptr`** will store the value in **`nullptr`**.
The variable named **`*ptr`** will store the value in **`nullptr`**.

None of these
None of these

Question 21 4 / 4 points

An array name is a pointer constant because the address stored in it cannot be changed at runtime.
An array name is a pointer constant because the address stored in it cannot be changed at runtime.

Question options:

True

False

Question 22 4 / 4 points

The following statement _____.

```
cin >> *num3;
```

The following statement _____.

```
cin >> *num3;
```

Question options:

stores the keyboard input in the variable **num3**

stores the keyboard input in the variable **num3**

stores the keyboard input into the pointer **num3**

stores the keyboard input into the pointer **num3**

is illegal in C++

is illegal in C++

stores the keyboard input into the variable pointed to by **num3**

stores the keyboard input into the variable pointed to by **num3**

None of these

None of these

Question 23 4 / 4 points

In C++11 you can use smart pointers to dynamically allocate memory and not worry about deleting the memory when you are finished using it.

In C++11 you can use smart pointers to dynamically allocate memory and not worry about deleting the memory when you are finished using it.

Question options:

True

False

Question 24 4 / 4 points

The ampersand (`&`) is used to dereference a pointer variable in C++.
The ampersand (`&`) is used to dereference a pointer variable in C++.
Question options:

True

False

Question 25 4 / 4 points

In C++11, the _____ keyword was introduced to represent address `0`.
In C++11, the _____ keyword was introduced to represent address `0`.
Question options:

`nullptr`
`nullptr`

`NULL`
`NULL`

`weak_ptr`
`weak_ptr`

`shared_ptr`
`shared_ptr`

None of these
None of these

Chapter 12

Question 1 4 / 4 points

The **`setprecision`** manipulator cannot be used to format data written to a file.

The **`setprecision`** manipulator cannot be used to format data written to a file.

True

True

False

False

Question 2 4 / 4 points

Data stored _____ disappears once the program stops running or the computer is powered down.

Data stored _____ disappears once the program stops running or the computer is powered down.

on a CD

on a CD

in RAM

in RAM

on a flash drive

on a flash drive

on the disk drive

on the disk drive

None of these

None of these

Question 3 4 / 4 points

File output may be formatted the same way as console screen output.

File output may be formatted the same way as console screen output.

True

True

False

False

Question 4 4 / 4 points

In order, the three-step process of using a file in a C++ program involves

In order, the three-step process of using a file in a C++ program involves

insert a disk, open the file, remove the disk

insert a disk, open the file, remove the disk

create the file contents, close the file, name the file

create the file contents, close the file, name the file

open the file, read/write/save data, close the file

open the file, read/write/save data, close the file

name the file, open the file, delete the file

name the file, open the file, delete the file

None of these

None of these

Question 5 4 / 4 points

When you store data in a variable, it is automatically saved in a file.

When you store data in a variable, it is automatically saved in a file.

True

True

False

False

Question 6 4 / 4 points

The **`ios::out`** flag causes the file's existing contents to be deleted if the file already exists.

The **`ios::out`** flag causes the file's existing contents to be deleted if the file already exists.

True

True

False

False

Question 7 0 / 4 points

If a file already exists, you can open it with the flags **`ios::in | ios::out`** to preserve its contents.

If a file already exists, you can open it with the flags **`ios::in | ios::out`** to preserve its contents.

True

True

False

False

Question 8 4 / 4 points

An alternative to using the **`open`** member function is to use the file stream object declaration itself to open the file. For example:

```
fstream DataFile("names.dat", ios::in | ios::out);
```

An alternative to using the **`open`** member function is to use the file stream object declaration itself to open the file. For example:

```
fstream DataFile("names.dat", ios::in | ios::out);
```

True

True

False

False

Question 9 4 / 4 points

When a file is opened, the file stream object's "read position" is

When a file is opened, the file stream object's "read position" is

at the end of the file

at the end of the file

at the beginning of the file

at the beginning of the file

nonexistent until the programmer declares it

nonexistent until the programmer declares it

in the middle of the file

in the middle of the file

None of these

None of these

Question 10 4 / 4 points

Which of the following access flags, when used by itself, causes a file's contents to be deleted if the file already exists?

Which of the following access flags, when used by itself, causes a file's contents to be deleted if the file already exists?

`ios::app`

`ios::app`

`ios::in`

`ios::in`

`ios::out`

`ios::out`

Any of these

Any of these

None of these

None of these

Question 11 4 / 4 points

To write to a file, you use the **`file_write`** function.

To write to a file, you use the **`file_write`** function.

True

True

False

False

Question 12 4 / 4 points

When data is read from a file, it is automatically stored in a variable.

When data is read from a file, it is automatically stored in a variable.

True

True

False

False

Question 13 4 / 4 points

When used by itself, the **`ios::app`** flag causes the file's existing contents to be deleted if the file already exists.

When used by itself, the **`ios::app`** flag causes the file's existing contents to be deleted if the file already exists.

True

True

False

False

Question 14 4 / 4 points

What is TRUE about the following statement?

```
out.open("values.dat", ios::app);
```

What is TRUE about the following statement?

```
out.open("values.dat", ios::app);
```

If the file already exists, its contents are preserved and all output is written to the end of the file.

If the file already exists, its contents are preserved and all output is written to the end of the file.

If the file already exists, it should be replaced with a new copy of **`values.dat`**.

If the file already exists, it should be replaced with a new copy of **`values.dat`**.

If the file already exists, it can be opened but not modified.

If the file already exists, it can be opened but not modified.

None of these

None of these

Question 15 Question this 0 / 4 points

All stream objects have _____ which indicate the position of the stream.

All stream objects have _____ which indicate the position of the stream.

error state bits

error state bits

condition statements

condition statements

markers

markers

intrinsic error messages

intrinsic error messages

← my answer

None of these

None of these

Question 16 4 / 4 points

By default, files are opened in binary mode.

By default, files are opened in binary mode.

True

True

False

False

Question 17 4 / 4 points

Outside of a C++ program, a file is identified by its _____while inside a C++ program, a file is identified by a(n) _____.

Outside of a C++ program, a file is identified by its _____while inside a C++ program, a file is identified by a(n) _____.

file number, file name

file number, file name

file name, file number

file name, file number

name, address

name, address

name, file stream object

name, file stream object

None of these

None of these

Question 18 4 / 4 points

To set up a file to perform I/O you must declare

To set up a file to perform I/O you must declare

at least one variable, the contents of which will be written to the file

at least one variable, the contents of which will be written to the file

one or more file stream objects

one or more file stream objects

a **`string`** object to store the file contents

a **`string`** object to store the file contents

All of these

All of these

None of these

None of these

Question 19 4 / 4 points

Closing a file causes any unsaved information still held in the file buffer to be

Closing a file causes any unsaved information still held in the file buffer to be

saved to the file

saved to the file

deleted

deleted

retained in the buffer for safekeeping

retained in the buffer for safekeeping

duplicated

duplicated

None of these

None of these

Question 20 4 / 4 points

Only one file stream object can be declared per C++ program.

Only one file stream object can be declared per C++ program.

True

True

False

False

Question 21 4 / 4 points

The **`ios::hardfail`** bit is set when an unrecoverable error occurs.

The **`ios::hardfail`** bit is set when an unrecoverable error occurs.

True

True

False

False

Question 22 4 / 4 points

The end-of-file marker is automatically written

The end-of-file marker is automatically written

when a file is opened with **`ios::eof`**

when a file is opened with **`ios::eof`**

when a file is opened with **`ios::app`**

when a file is opened with **`ios::app`**

when a file is closed

when a file is closed

when the program ends

when the program ends

None of these

None of these

Question 23 4 / 4 points

When passing a file stream object to a function, you should always pass it by reference.

When passing a file stream object to a function, you should always pass it by reference.

True

True

False

False

Question 24 4 / 4 points

The _____ marker is the character that marks the end of a file and is automatically written when the file is closed.

The _____ marker is the character that marks the end of a file and is automatically written when the file is closed.

End Of File (**EOF**)

End Of File (**EOF**)

No More Data (**NMD**)

No More Data (**NMD**)

Data Stream Close (**DSC**)

Data Stream Close (**DSC**)

Data Read Stop (**DRS**)

Data Read Stop (**DRS**)

None of these

None of these

Question 25 Question this 0 / 4 points

Which of the following data types can be used to create files and read information from them into memory?

Which of the following data types can be used to create files and read information from them into memory?

ofstream

ofstream

istream

istream

ifstream

ifstream

instream

instream

None of these

None of these

← my answer, only fstream can do this