Question 1

1 / 1 point

A _____ function is automatically called when an object is first created in memory.

- destructor
- ✓ constructor
- modifier
- accessor

Question 2

0 / 1 point

It is possible for a class to have multiple constructor functions as long as each has a unique parameter list.

- ⇨ True
- ✗ False

Question 3

1 / 1 point

In C++, the class definition is typically stored in a header file with extension _____.

- .cpp
- ✓ .h
- .header
- .txt

Question 4 1 / 1 point

The **Temperature** class definition is listed below. In the **main** function, which code is invalid?

```
class Temperature {
private:
  double temp;          // Fahrenheit temperature
public:
  Temperature( ) { temp = 0.0; }
  double getTemp( ) { return temp; }
  void setTemp (double t) { temp = t; }
 };
```

- [✓] Temperature winter;
  cout << winter.temp;
- [ ] Temperature winter;
  cout << winter.getTemp( );
- [ ] Temperature winter;
  winter.setTemp(32.0);
- [ ] All of the examples are legal

Question 5 1 / 1 point

The **Temperature** class definition is listed below. Which statement is true about the code in the **main** function?

```
class Temperature {
private:
  double temp;          // Fahrenheit temperature
public:
  Temperature( ) { temp = 0.0; }
  double getTemp( ) { return temp; }
  void setTemp (double t) { temp = t; }
 };

// code in main
Temperature winter;
if (winter.getTemp( ) == 0.0) cout << "It is cold";
else cout << "Unknown";
```

- ✓ ○ The output is: It is cold
- ○ The output is: Unknown
- ○ The result is unpredictable since the private data stores an undefined value
- ○ An error occurs because the private data has not been initialized

Question 6 — 1 / 1 point

An application file (ex. Source.cpp) wishes to create objects of the user-defined **Temperature** class. Which line should be added to the top of the application?

- ○ #include <Temperature.h>
- ✓ ○ #include "Temperature.h"
- ○ #include "Temperature"
- ○ #pragma once

Question 7 — 1 / 1 point

In a C++ **struct** data members are automatically _____ while in a C++ **class** data members are typically _____.

- ○ public, public
- ○ private, public
- ✓ ○ public, private

◯ private, private

Question 8 1 / 1 point

In object-oriented programming, an class should protect an object's data by making its visibility private and by providing public methods to edit the data in controlled ways.

[✓ True]
◯ False

Question 9 1 / 1 point

Using class **Temperature**, what is the result of the code in the **main** function?

```
class Temperature {
private:
  double temp;                  // Fahreinheit temperature
public:
  Temperature( );               // POST: temp is set to 0
  Temperature (double t); // POST: temp is set to t
  double getTemp( );            // POST: return temp
  void setTemp (double t);// POST: temp is set to t
};

// in main function
Temperature spring(50.0);
cout << spring.getTemp( );
```

◯ 0.0 displays

[✓ 50.0 displays]

◯ an undefined value displays

◯ an error occurs as the main function cannot access private data

Question 10 — 1 / 1 point

Accessor methods in a class often use a name beginning with ______.

- ○ set
- ○ the class name
- ✓ ○ get
- ○ public

Question 11 — 1 / 1 point

The following two methods would be considered **overloaded methods** since they both return **double** values and have a single parameter of type **double**.

public double mysteryOne (double m) { ... }

public double mysteryTwo (double m) { ... }

- ○ True
- ✓ ○ False

Question 12 — 0 / 1 point

A **Circle** class uses a private data member named **radius** to store a radius (ex. 4.5). The function heading for the accessor method looks as follows. What is the meaning of **const**?

double getRadius ( ) const
{ return radius: }

- ○ The value returned by this function must always be stored in a named constant.
- ✗ ○ A **Circle** object is never allowed to alter the **radius** data member, even via the public modifier function.
- ⇨ ○ This function is guaranteed not to alter an object's private **radius** data.

- ○ This function can only be called on a constant **Circle** object, one that is declared and initialized in user code but cannot be subsequently modified.

Question 13 — 0 / 1 point

Which statement is not true about objects of the C++ string class?

- ○ A C++ string object can be constructed from either another C++ string or a C string
- ✗ ○ C++ strings objects permit access to individual characters via the subscript operator [ ]
- ○ Use of the relational operators is permitted when comparing two C++ strings **s1** and **s2**, as in: **if (s1 == s2) cout << "same contents";**
- ⇨ ○ All statements a), b) and c) are true about C++ strings

Question 14 — 1 / 1 point

A class **Temperature** is created with a private data member named **temp**. Which declaration in the **main** function calls the default constructor?

- ○ Temperature weekday { temp };
- ○ Temperature today ( );
- ✓ ○ Temperature daily;
- ○ Temperature monday = 0;

Question 15 — 1 / 1 point

What is the output of the code?

```
string s = "Programming";
cout << s.substr(1,3);
```

- ○ ro
- ○ rg
- ✓ ○ rog

◯ Pro

Question 16 1 / 1 point

When working with input files, use the _____ preprocessor directive at the top of the program.

◯ #include <iostream>

◯ #include <string>

◯ #include <file>

✓◯ #include <fstream>

Question 17 1 / 1 point

The data file contains: 10 20 30 What is the output of the code? Assume the file opened correctly.

```
ifstream fin;
fin.open ("data.txt");
int num;
int sum = 0;
fin >> num;
while (! fin.fail( ))
{      cout << num << " ";
       sum = sum + num;
       fin >> num;
}
cout << sum;
```

◯ 10 20 30 30 90

◯ 30 30 30 120

- ✓ ○ 10 20 30 60
- ○ 10 10 20 30 30 60

Question 18 1 / 1 point

An attempt to read the end-of-file marker at the end of a data file will cause the file stream to go into fail state

- ✓ ○ True
- ○ False

Question 19 1 / 1 point

Which code segment correctly establishes an input string stream from the contents of string line and reads the integer and double values?

string line = "55 7.99";

○
```
istringstream is (line);
int k;
double d;
line >> k >> d;
```

✓ ○
```
istringstream is (line);
int k;
double d;
is >> k >> d;
```

○
```
istringstream is (line);
int k;
double d;
getline(is, k);
getline(is, d);
```

○
```
istringstream is (cin);
int k;
double d;
cin >> k >> d;
```

Question 20 1 / 1 point

Use the _____ function to convert a string of digits (ex. "1234") to an integer for storage in an **int** variable.

- ○ to_string
- ○ stod
- ✓ ○ stoi
- ○ int