Exam 1, Fall 2018 **Name**___________________________________

MC: each one point
(write question number followed by the answer)

1. C++ provides a number of features that "spruce up" the C language, but more importantly, it provides capabilities for ________ that were inspired by the Simula simulation programming language.
a. structured programming
b. simulation
c. object-oriented programming
d. None of the above.

2. Which statement is *false*?
a. Classes are reusable software components.
b. A class is to an object as a blueprint is to a house.
c. Performing a task in a program requires a method.
d. A class is an instance of its object.

3. Which of the following does *not* cause a syntax error to be reported by the C++ compiler?
a. Mismatched `{}`.
b. Missing `*/` in a comment.
c. Missing `;` **at the end of a statement**.
d. Extra blank lines.

4. Which of the following is *not* a valid C++ identifier?
a. `my Value`
b. `_AAA1`
c. `width`
d. `m_x`

5. The `std::endl` stream manipulator________.
a. inputs a newline.
b. flushes the output buffer.
c. outputs a newline and flushes the output buffer.
d. terminates the program.

6. What is the value of `result` after the following C++ statements execute?

```
        int a{4};
        int b{12};
        int c{37};
        int d{51};
        int result{d % a * c + a % b +
a};
```

a. 119
b. 51
c. 127
d. 59

7. Each class you create becomes a new ________ you can use to declare variables and create objects.
a. variable
b. object
c. type
d. access modifier

8. Typically, you cannot call a member function of a class until you create a(n) ________ of that class.
a. object
b. image
c. header
d. constructor

9. To call a member function for a specific object, you specify the object's name, followed by a(n) ________, then the member function name and a set of parentheses.
a. dot operator
b. colon
c. `::`
d. `->`

10. How many times will the following loop print `hello`?

```
    i = 1;

    while (i <= 10) {
       cout << "hello";
    }
```

a. 0.
b. 9.
c. 10.
d. An infinite number of times.

11. Assuming that `x` is equal to `4`, which of the following statements will *not* result in `y` containing the value `5` after execution?
a. `y = 5;`
b. `y = x++;`
c. `y = ++x;`
d. `y = x + 1`

12. The OR (`||`) operator:
a. Has higher precedence than the AND (`&&`) operator.
b. Stops evaluation upon finding one condition to be `true`.
c. Associates from right to left.
d. Is a ternary operator.

13. Consider the following code, assuming that `x` is an `int` with an initial value of `12`

```
    if(x = 6) {
       cout << x;
    }
```

What is the output?
a. 6
b. 12
c. Nothing.
d. A syntax error is produced.

14. The function prototype

```
        double mySqrt(int x);
```

a. Declares a function called `mySqrt` which takes an integer as an argument and returns a `double`.
b. Defines a function called `double` which calculates square roots.
c. Defines a function called `mySqrt` which takes an argument of type `x` and returns a `double`.
d. Declares a function called `mySqrt` which takes a `double` as an argument and returns an integer.

15. A function prototype does *not* have to:
a. Include parameter names.
b. Terminate with a semicolon.
c. Agree with the function definition.
d. Match with all calls to the function.

16. Which of the following *is not* a valid enumeration?
a. `enum class Person {ME, YOU, THEM};`.
b. `enum class Person {ME = 1, YOU = 2, THEM = 3};`.
c. `enum class Person {ME = 0, YOU = 0, THEM = 0};`.
d. `enum class Person {ME, YOU, ME};`.

17. When an argument is passed-by-value, changes in the called function __________ affect the original variable's value; when an argument is passed call-by-reference, changes in the called function __________ affect the original variable's value.

a. Do not, do.

b. Do not, do not.

c. Do, do.

d. Do, do not.

18. Call-by-reference can achieve the security of call-by-value when:

a. The value being passed is small.

b. A large argument is passed in order to improve performance.

c. A pointer to the argument is used.

d. The `const` qualifier is used.

19. Which statement would be used to declare a 12-element integer `array` c?

a. `array c<12>;`

b. `array c<int, 12>;`

c. `array<12> c;`

d. `array<int, 12> c;`

20. Assume that the `array` named `items` contains the integer values 0, 2, 4, 6 and 8. Which of the following set of statements uses the range-based `for` loop to display each value in `items`?

a.
```
for (int i = 0; i <items.size();
++i) {
   cout <<items[i] <<endl;
}
```

b.
```
for (int item : items) {
   cout <<items[item] <<endl;
}
```

c.
```
for (int item : items) {
   cout <<item <<endl;
}
```

d.
```
for (int item : items.size()){
   cout <<item <<endl;
}
```

Programs

1. **(10 points)** Write a **function template** called *maximum* that can determine the largest of three arguments. Note the argument can be int, double, or char**.**
2. **(20 points)** Write a **complete C++ program** (include all required headers if there is any) to simulate of generating poker card from a deck of 52 cards. A poker deck contains cards with Suit

   "Hearts", "Diamonds", "Spades", "Clubs" and Face "Ace", "Two", "Three", "Four", "Five", "Six", "Seven", "Eight", "Nine", "Ten", "Jack", "Queen", "King".

   Your program will randomly generate five cards and output them on the screen. You will use C++ default_random_engine and uniform_int_distribution classes to generate random numbers which can map to the poker cards in our application. To simplify your output, you may use 1 for "Ace", 2 for "Two", …13 for "King". The following is an example of your output: H1, H7, D3, C10, and C12 if your program randomly generate Heart Ace, Heart Seven, Diamond Three, Club Ten, and Club Queen.

   Note all 52 cards are unique so your program cannot generate cards with same suit and face value.
3. **(15 points)** Write a **complete C++ program** to ask user to enter two positive numbers, your program will output the sum from the smaller number to the larger number inclusive. For example, if user entered 10 and 1 then you will compute sum from 1 to 10 which is 55. Your program will output the sum and the sum/100. Because the sum/100 may give us fraction, so your program will the output of sum in integer format but the sum/100 which contains 3 decimal digits. In the example above,

   The sum of 1 to 10 is 55, and 55/100 is 0.550

   If user entered 1 and 100, then the sum is 5050 and the percent is 50.500. If user entered invalid value then your program will continue to ask the user until user enters positive number. You may assume user will enter 0, positive number, and negative number only. Note 0 or negative numbers are considered as invalid numbers in our program computation.
4. **(10 points)** Convert the following C++ program from *for* loop to a *while* loop
   ```
   for (int k=2; k <=20; k+=3) {
     if (k == 3) continue;
     cout << k <<", " << k*k << endl;
   }
   ```
5. **(15 points)** Write a C++ class call **Account**. An Account object can be described by using the following: first name of owner, last name of owner, address of owner, social security number, current balance, Write setters and getters of all these attributes. This class contains at least one constructor. You may give more constructors if you like. In addition to the setters and getters, this class contains a member function call it *toString*. The *toString* member function will output the message about the Account object. I.e., it shall output, the names of the owner of this account, owner's address, owner's social security number and the current balance.
6. **(10 points)** (a) Write a recursive function to compute factorial function. (b) write a iterative function to compute factorial function.