CMPS-2020 Final Exam Spring 2021 name ___________________________

To take this exam, create a file on Odin named:

`2020/f/final.txt`

You may start with the given text file on our web page.
Clearly indicate your answers in the text file.

Object Oriented
----

1. A class which dynamically allocates memory to its pointer data members...
   (choose 2 correct answers)
      A. will have all public members
      B. needs a destructor
      C. needs no copy constructor
      D. should have an overloaded assignment operator

2. Indicate the name of the base class in the following statement:

```
class Person : public Student
```

3. Below is a function that overloads an operator.
   Fill in the blank to give the correct return type to this function.

```
________ operator <= (const Number &v) { return (value <= v.value); }
```

Linked Lists
----

1. T F A programmer must know in advance how many nodes will be needed in a linked list.

2. T F A circular linked list requires no additional elements be added to the Node structure.

3. T F The nodes of a linked list are stored in contiguous memory, similar to an array.

4. T F When the head pointer is null or nullptr, it signifies an empty list.

5.
Write a function to add a node to a singly-linked list. Near the beginning of your program, the head node was declared like this:

```
Node *head = null_ptr;
```

Assume a node structure similar to our lab and homework. Look closely and write a perfect function. A call to your function looks like this:

```
head = addNode(head, 35);
```

---write your function here---

Binary Search Trees

You are going to insert some data values into a binary search tree.
Your data values are the numbers 1 through 9. Your tree starts out empty.

1. Show the order in which you would insert the data into the tree, that would make the tree search function work inefficiently. Show the order now.

2. Now show an order of data insertion that will make your tree work efficiently. There can be more than one correct answer to this question. Remember that a tree works best when it is balanced.
Show your order now.

You should draw your tree from question #2 above. Use characters something like this.
In place of the x characters, show your data values.

```
      x
     / \
    x   x
   / \
  x   x
```

3. Now answer the following questions for the tree you constructed in question #2:

    a. How many levels does your tree have? ____
    b. How many nodes have exactly one child? ____
    c. List the data values using pre-order traversal:
    d. List the data values using in-order traversal:
    e. List the data values using post-order traversal:

Hash Tables
----

1. How do you calculate a hash table's load factor?

2. Why do you need to know a hash table's load factor?

3. What is the job of a hash function?

4. An empty hash table is below with size 7 and hash function: **hash(x) = (2x + 5) mod size**.
   note: x is your data value.

    Write a complete hash function using C++ here:

5. Using the hash function from #4, insert the following keys into the table,
   in this order. 3, 4, 7, 11, 8, 10

   Use linear probing to resolve collisions.
   Do not resize the table.

```
+------+------+------+------+------+------+------+
|      |      |      |      |      |      |      |
+------+------+------+------+------+------+------+
   0      1      2      3      4      5       6
```

Stacks and queues
----

1. A program using a stack data structure should never attempt to pop from the stack before pushing something to the stack. What action should your program take if it attempts to pop from a stack with no data items on the stack?

2. A <u>circular queue</u> data structure is good because it can never fill up. Is this true or false?

Big O Analysis
----

1. What is the Big-O efficiency of the following function, in terms of n?

```
int find_sum(int *arr, int n) {
    int sum = 0;
    for (int i=0; i<n; i++)
        for (int j=0; j<n; j++)
            sum = sum + arr[i] + arr[j];
    return sum;
}
```

Write your answer here: _____________

2. Reduce the following expression asymptotically and show using Big-O notation.

$n/2 + \log n + 10n$ ______________________________

3. Name two algorithms we learned that use a divide-and-conquer or reduce-and-conquer strategy.

__________________________ __________________________

General
----

1. A boolean expression will evaluate to true or false. Here is a sample boolean expression: (a > b) Now, write a boolean expression that will evaluate to true if a variable named **x** is greater than 4 but not greater than 24.

2. Below is a function to allocate memory. A pointer is passed in as an argument. After the function call, you want the pointer argument to contain a valid memory address.

```
void get_memory(int *ptr, int size) {
     ptr = new int[size];
}
```

This function will compile and run, but it will not operate correctly.

a. Explain what is wrong with this function.

b. Show a way that you would correct the function.

3. The ordered array below is in your program. Write some code that will remove data value 8 from the array. Do this by shifting array elements to the left starting at value 9. Shift the data elements inside a for-loop. Remember to maintain the integrity of your list and your program. Only 3-lines of code are needed.

```
int arr[] = {2, 4, 6, 8, 9, 10, 12, 13, 15, 16, 21, 28};
int n = 12;
```

4. Complete the code below to initialize num with a random value that falls within the range -5 to 5. No additional lines of code are needed.

```
int num = rand()
```

5. What is the output of the following code segment? Briefly explain what each line of code does.

```
char a[] = "Sum of errors!";
a[3] = a[2];
a[4] = a[7];
a[5] = a[8];
a[6] = a[strlen(a)];
cout << a;
```