NYU, Tandon School of Engineering
Bridge to Computer Science Program — Summer 2022 Extended

Third Midterm Exam

- There are 100 points total.
- Note that there are longer programming problems at the end. Be sure to allow enough time for these.
- We supplied you with a file, named ‘solutions.txt’, where you should type all your answers in.
- For editing this file, you are allowed to use compilers such as Visual Studio, VSCode, XCODE, CLion, textedit and notepad
- You may use 2 scratch papers.
- Calculators are not allowed.
- This is a closed-book exam. No additional resourced are allowed.
- Pay special attention to the style of your code. Indent your code correctly, choose meaningful names for your variables, define constants where needed, choose most suitable control statements, etc.
- In all questions you may assume that the users enter inputs as they are asked. For example, if the program expects a positive integer, you may assume that users will enter positive integers.
- No need to document your code in this exam, but you may add comments if you think they are needed for clarity.
- Read every question completely before answering it.
- When done, **submit on Brightspace, Gradescope AND email to dkatz@nyu.edu**

1. (4 pts) Given the code:

```
int* ptr = new int(100);
int* other = ptr;
cout << *other << endl;
delete other;
delete ptr;
```

what common coding error have we made in this code fragment?

   a. memory leak
   b. double delete
   c. dereferencing NULL
   d. none of the above; or no error.

2. (4 pts) What is wrong with the following code ?

```
class Puzzle {
        virtual void mixItUp()=0;
};
int main() {
        Puzzle* p;
        p->mixItUp();
}
```

3. (4 pts) Provide the function signature (just the prototype; do not implement the function) for a "reverse" function which could be placed inside the Queue class in order to return a stack that is a copy of the existing stack but reversed in order.

4. (4pts) Convert the following in-fix expression to postfix form: 2+3+4*(6-5)

5. (4 pts) Given a pointer "ptr" which points to the first item in an array of integers which is created on the heap, write a single line of code which will release the memory allocated to that array.

6. (5 pts) We've created a class called Thing which contains a function called "callMe." You are given a pointer called ptr which points to somewhere in the middle of an array (not vector) of Things. Please provide a single line of code which will call the "callMe" function for the element **after** the one pointed to by ptr.

7. (10 pts) Please provide the overloaded **unary** operator- (the negation operator) for the stack class. The “negation” of a stack returns a stack where the order of the elements it contains is reversed. You may overload this as a member and may assume a stack class similar to the one we defined in our webinar.

8. (10 pts) You will be given an unsorted vector of integers and asked to find the number of unique (non-repeating) elements in the vector. The vector is very large and sorting it is not an option because the elements must remain in their original position and you cannot copy the vector. Describe a way, **in English**, how you could determine the number of unique items in the vector in O(N log N) time, or faster, while using a minimum of additional space.

9. (15 pts) An ellipsis is an indication that information is omitted from text. It is, usually, represented as three periods somewhere in text. For example, “She opened the door…and…she saw … her friend!” Sometimes the ellipsis is immediately adjacent to the text and sometimes it is separated from text with a space but the three period characters will always be consecutive.

   Write a program to ask the user for the name of a file and print the number of ellipses (that’s the plural of ellipsis; yes I had to look it up) that exist inside the file. You do not have to worry about “fringe” cases such as if the ellipsis begins on one line and ends on another.

10. (20 pts) You are given a pointer to the root node of a BinaryTree of Integers. For this problem, you are asked to write a member function “is_height_tree” which determines if the data value of every node is equal to the height of that node. Your function should return true or false and must run in linear time.

11. (20 pts) Given a vector of integers, we would like to write a function called “strangeSort” to sort the integers such that all even values are in the first, in their original order, followed by all odd values in the reverse of their original order. (zero will not be present in this vector). Your solution should be as time efficient as possible but you may use as much additional space as you’d like. See the below for some examples

Input: [1,2,3,4] Result: [2,4,3,1]	Input: [1, 7, 3, 4, 2, 8] Result: [4, 2, 8, 3, 7, 1]
Input: [3, 5, 7, 6, -2, 8] Result: [6, -2, 8, 7, 5 ,3]	Input: [2, 4] Result: [2, 4]
Input: [7, 2, 3, 4, 9] Result: [2, 4, 9, 3, 7]	Input: [] Result: []