

1. If you want to get dynamic memory from the computer during run-time, you need to use which system call?
 - a. new
 - b. delete
 - c. memcpy
 - d. strcpy
2. To return the memory you got dynamically at run-time to the system, which function would you call?
 - a. new
 - b. delete
 - c. memcpy
 - d. strcpy
3. To get an array of dynamic memory, what would the function call look like?
 - a. new Array[10];
 - b. delete [] Array;
 - c. memcpy Array[10];
 - d. strcpy Array[10];
4. To return the dynamic array to the system what would the function call look like?
 - a. new Array[10];
 - b. delete [] Array;
 - c. memcpy Array[10];
 - d. strcpy Array[10];
5. When you have a class with dynamic data, which of the follow methods should not always be implemented?
 - a. copy constructor
 - b. parameterized constructor
 - c. assignment operator
 - d. destructor
6. What invokes the assignment operator?
 - a. MyList = OldList;
 - b. Stack S = OldStack;
 - c. Passing an object with dynamic memory into a function.
 - d. Creating an object.
7. What does not invoke the copy constructor?
 - a. MyList = OldList;
 - b. Stack S = OldStack;
 - c. Passing an object with dynamic memory into a function.
 - d. Returning an object from a function.

8. When is the destructor invoked?
 - a. At the end of a function.
 - b. At the end of a nested scope
 - c. When ever the object goes out of scope
 - d. All of the above

9. Where does data pushed in a stack?
 - a. Bottom
 - b. Front
 - c. Rear
 - d. Top

10. Where does data pop from in a stack?
 - a. Bottom
 - b. Front
 - c. Rear
 - d. Top

11. Where does data enqueue in a queue?
 - a. Bottom
 - b. Front
 - c. Rear
 - d. Top

12. Where does data deque from a queue?
 - a. Bottom
 - b. Front
 - c. Rear
 - d. Top

13. Does it really matter how you implement stacks and queues as long as they behave like a stack and a queue?
 - a. Yes
 - b. No

14. Which parameter of main tells you how many command line arguments were passed to main?
 - a. Huh?
 - b. Argv
 - c. Argc
 - d. What?

15. Which parameter of main is an array of the command line argument which were passed to main?
 - a. Huh?
 - b. Argv
 - c. Argc
 - d. What?

16. How would you open a file whose name is the 2nd argument passed to main?
- `in.open(Argv[2].c_str());`
 - `in.open(Argv[1].c_str());`
 - `in.open(Argv[2]);`
 - `in.open(Argv[1]);`
17. What is always the first element of argv?
- The command file
 - A data file
 - The name of the executable
 - The location of the executable
18. What is the order of steps for insertion in a linked list, assuming insertion after where the current pointer points?
- Have the current node's next pointer point to the new node.
Have the new node's next pointer point to where the current node's next pointer points to.
 - Create a new node.
Have the current node's next pointer point to the new node.
 - Create a new node.
Have the new node's next pointer point to where the current node's next pointer points to.
Have the current node's next pointer point to the new node.
 - Create a new node.
Have the new node's next pointer point to where the current node's next pointer points to.
Have the current node's next pointer point to the new node.
19. What is the order of steps for deletion in a linked list, assuming you are deleting where the current pointer points?
- Delete the current node
 - Find the node before the current node
Delete the current node
Have the node before current next pointer to where the current node's next pointer points.
 - Find the node before the current node
Delete the current node
Have the node before current next pointer to where the current node's next pointer points.
 - Find the node before the current node
Have the node before current next pointer to where the current node's next pointer points.
Delete the current node.
Make the current node be the node before the old current node
20. I can't wait to program a linked-list.
- Yes!!