1. What is the role of a prototype program in problem solving?

To lay out the basics and give you a starting point in the actual problem solving.

2. What stages in the software life cycle are influenced by the testing stage?

Elaboration, Construction, Transition

3. What are the main advantages associated with object-oriented programming?

Using different modules allows for easier debugging

4. Where do C++ programs begin to execute?

At the root

5. What is a variable?

a value/word that can assume any of a set of values

6. Where are variables declared in a C++ program?

at the top

7. What is the main difference between a while and a do...while statement?

a while statement will only process if the statement is met, while a do...while will always process once, then only continue if the statement is met.

8. What is typically included in a class definition?

the keyword class followed by they class name, on the inside you declare public and private declarations of your class

9. What is the difference between a data member and a local variable inside a member function?

A variable inside a member function can only be accessed inside that member and a data member can be accessed throughought the program.

10. What is the difference between a constructor and a function?

A constructor is a function used to initialize an object's data when it is created.

11. When does C++ create a default constructor?

In any class that does not explicitly include a constructor.

12. How many constructors can be created for a class?

by default just one, but they may be overloaded to create as many constructors as necessary

13. What is the difference between a function prototype and a function definition?

Function prototype is located in the .h file and only contains the access function name and paramater type. Function definition contains the code for the function to perform its activity.

14. What is the role of a header-file?

the header file shows the user what types go into and come out of a function but does not allow a user to manipulate the actual code for the function.

15. What does a function signature include?

Name, paramaters, scope, and other general function information

16. What is the scope of global variables?

The scope of global variables are created by placing variable declarations outside any class or function definition. Global variables retain their values throughout the execution of the program and they can be referenced by any function that follows their declarations or definitions in the source file.

17. How does the compiler handle inline functions?

When the compiler inline-expands a function call, the function's code gets inserted into the caller's code stream (conceptually similar to what happens with a #define macro). This can, depending on a zillion other things, improve performance, because the optimizer can procedurally integrate the called code — optimize the called code into the caller.

18. How does the compiler handle inline functions?

When the compiler inline-expands a function call, the function's code gets inserted into the caller's code stream (conceptually similar to what happens with a #define macro). This can, depending on a zillion other things, improve performance, because the optimizer can procedurally integrate the called code — optimize the called code into the caller.

19. What is the main advantage associated with function arguments that are passed by reference?

if you pass by reference, you can modify the value as opposed to passing by value where you cannot change the value

20. How are overloaded functions differentiated by the compiler?

paremeters

21. When defining a recursive function, what are possible causes for infinite recursion?

Not having a base case, or building a recursion process that doesn't converge on the base case.

22. What are the similarities between iteration and recursion?

they are both based on a control statement, both involve repetition, both involve a termination case, both graduatlly approach that termination case, and both can occur infinitely

23. What are the two different ways of specifying the length of an array?

within the brackets or by using an initializer list.

24. What is the main difference between strings declared using the type string versus strings declared using an array of characters?

array of characters need a termination character as well as size specification whether its explicit or implicit.

25. How are arrays passed to functions?

The function recieves both the array and the array size as parameters. function(arrayName, arraySize)

26. What is the difference between an array declared as static, and one that is not?

static arrays keep the values after the end of a function, while non static reinitialize every time.

27. How many dimensions need to be specified when passing a multi-dimensional array as an argument to a function?

as many as there are dimensions. Most commonly just rows and columns

28. In one sentence, what is the main idea implemented by insertion sort?

it goes through the list only once, picking each integer and putting it in its desired position, then continuing.

29. In one sentence, what is the main idea implemented by selection sort?

Places the smallest item in the list at position 1, and then proceeds to each value until the last position of the ray is reached.

30. What is the number of operations for insertion sort under a best-case scenario, and what is the best-case scenario?

n-1 The best case is when the array is already sorted.

31. What is the base case for a recursive implementation of merge sort?

if the array length is less than or equal to 1, then that array is returned to the other array and merged together

32. What is a pointer?

a variable that contains the address of another variable

33. What does the address (&) operator return?

the address of the variable it is attached to

34. What does the star (*) operator return?

will return zero or many characters placed before the operator

35. What does the sizeof operator return?

Returns size of operand in bytes.

36. What are the different ways to pass a pointer to a function?

you can pass a pointer to a pointer or pass a reference to pointer.

37. What is a function pointer?

Function Pointers are pointers, i.e. variables, which point to the address of a function.

38. What is a linked list?

A linked list is a data structure that is not necessarily in the same contiguous memory space (such as arrays). It holds the data type and points to the next data item in the list, or in a doubly linked list also to the previous item.

39. What is the main advantage of linked lists over arrays?

Nearly infinite size, limited only by system memory and also the ability to expand the size dynamically.

40. What is the main advantage of arrays over linked lists?

You can access specific elements quickly in an array. In a linked list you must traverse the list to get access the same element.

41. How are linked lists passed as arguments to a function?

Pass the head pointer to a function as a reference argument

42. What is the difference between a circular linked list and a basic linked list?

in a circular linked list, the last node contains a pointer that goes back to the first node; in a basic linked list, the last node contains a null pointer

43. What is the main disadvantage of a doubly-linked list over a basic linked list?

they take up twice as much memory for each node

44. What is a stack?

An ADT whose most recently inserted item is the first item removed or retrieved. This property is called last in, first out (LIFO). Items enter and leave a stack at its top.

45. What are the two main functions defined by a stack?

push (insert an item) pop (remove an item)

46. How can you implement a stack with an array?

Declare a variable that points the last item in the array, which increases (++) when items are pushed onto the stack, and decreases (--) when an item is popped off the top.

47. How can you implement a stack with a list?

A singly-linked list is already a lot like a stack, it just depends on the way you address moving the head.

48. How are infix expressions evaluated by computers?

they use parenthisis and a stack to determine which evaluations need to be made first

49. What operations would you need to perform to find a given element on a stack?

either traverse the entire list and pop the given part or create a pointer system that automatically points to it.

50. What is a queue?

Stores a set of elements in a particular order with a First In, First Out principle

51. What are the two main functions defined by a queue?

The two functions are, Enqueue and Dequeue

52. How can you implement a queue with an array?

by keeping track of a front item and a rear item, or with a circular array that uses the modulo function

53. How can you implement a queue with a list?

Implement the operation dequeue() as the list operation remove(1) and the operation getFront(queueFront) as the list operation retrieve(1, queueFront)

54. What is the stack operation corresponding to the enqueue operation in queues?

Push

55. What is the height of a tree?

The height of a tree is in terms of the levels of its nodes such that, if the tree is empty its height is 0 otherwise its height is equal to the maximum level of its nodes.

56. What is a leaf?

A new item to the tree--a child

57. What is a binary tree?

a tree where each parent can have at max two children.

58. What is a binary search tree?

it is similar to a binary tree. The keys in a nonempty left subtree (right subtree) are smaller (larger) than the key in the root of subtree. the values in all nodes in the left subtree of a node are less than the node value the values in all nodes in the right subtree of a node are greater than the node values

59. What is the inorder traversal of a binary tree?

you would go to the furthest down left most node, then to the root then to the right (if left and right exist) then you would return one node previous and do the same until you reached the root, then go to the furthest down left most node on the right side of the root and continue this process

60. What are the elements typically included in a class definition?

The elements typically included in a class definition are the function prototypes, usually declared public, and the data members used in the class, which are usually declared private.

61. What are the access-specifiers that can be used in a C++ class definition?

Public, Private, Restricted

62. How are objects initialized when they are created?

ie. GradeBook myGradeBook(); call upon the class and then give it a name like shown

63. How are objects initialized when they are created?

obj. are initialized by the variables that call them, when they are created within a class. e.i. classname objectname etc.

64. What is a function signature?

It is what uniquely separates overloaded functions. It includes the return type, and the parameter list of the function.

65. What is a recursive function?

a recursive function is a function that during execution, calls itself over and over until a base case is reached

66. What is the alternative way to solve a problem that could be solved through recursive functions?

Iteration by using loops

67. What is the difference between an array that is declared as static and one that is not?

static arrays can be called from anywhere

68. What is the main difference between a string of characters that is read into a variable of type string versus a variable of type char[]?

In arrays strings are stored as a characters (char[]), each character of string will be stored in each memory location of the array, while string of characters is only one memory location.

69. Briefly describe the divide-and-conquer paradigm.

dividing and conquering means breaking a problem into two smaller problems, solving both of those, and putting the two solutions together.

70. Briefly describe in one sentence how does merge sort work?

merge sort divides the data into halves until data of one element is reached and then merges each element together according to its placement in comparison to the rest of the data

71. What is a pointer?

A pointer is a variable that points to an address in memory, which contains some data. The pointer does not contain or point to any data, only a memory address.

72. What is the experimental approach for measuring the running time of an algorithm?

run the code for n-times and get average values, drop the constant and lowest number. for example if f(x) = 3n +1 the running time will bef f(x) = O(n)

73. Briefly, how does selection sort work?

it selects the smallest element in a list and switches it with the element in its correct position, then it selects the next smallest and does the same...

74. What is the advantage of linked lists over arrays?

Linked lists can be dynamically allocated and their size can change as needed.

75. What is a queue?

a queue is a list of data that follows the fifo principle, an example of this would be when you get into a line at a movie theater...the first one there gets to buy a ticket first

76. What are the main operations associated with a stack?

push - Add an element to the top of the stack pop - remove the top element from the stack

77. What is the Euler tour traversal of a tree?

The Euler traversal through the tree in a rubber band style shape.

78. How many steps does it take to search a node in a binary search tree?

Log(n) where n is the number of nodes.