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

Prototype programming is an approach to programming that enables one to take an organized approach to developing an effective program with minimal errors and a strategic pattern when solving a problem. i.e. book gave an example of a costumer withdrawing money from a bank, the approach that was taking on a pseudo code level during the OOA/Design lvl before proceeding into creating a solution.

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

The Individual components and steps such as analysis, design, and implementing code as well as the entire system are tested for execution of the requirements identified during the analysis stage. i.e. Main output

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

Well for one encapsulation the values of the variables inside an object are private, unless methods are written to pass info outside of the object. As well as Inheritance where each subclass inherits all variables and methods of its super class. Example in the book included obj clock and how obj alarm would still use clock from the first class.

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

Within the main function

5. What is a variable?

Is a method or identifier I would say, we use to bind a data object to memory location; which is then stored in a location that can be accessed when and manipulated later when the variable name is called.

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

It depends if it’s a global then they have to be declared out side the source code to be used in every scope however a local variable is one declared in a local function etc. which obviously doesn’t need to be declared outside the variable seeing how it is used for the function or block its being called for.

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

)) What is the main difference between a while and a do...while statement? The do while construct consists of a block of code and a condition. First, the code within the block is executed, and then the condition is evaluated, this is done until it is proven false. The difference between the While loop is it tests the condition before the code within the block is executed.

8. What is typically included in a class definition?

class variables, function definitions for the class

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

local variables are accessed inside member functions only while data members can be accessed throughout the class

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

A constructor initializes an object. A function is part of an object.

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

when the programmer does not explicitly include a constructor

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

one

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

a function prototype simply declares the functions parameters, the function definition includes any necessary variables and the function's actual code

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

A header file declares a class, its data, and functions available through that class.

15. What does a function signature include?

A function signature includes a function's return type, parameters and semi-colon.

16. What is the scope of global variables?

globally, anywhere in the program.

17. How does the compiler handle inline functions?

not answered

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

If a function's arguments are large in size, computing time and memory space is not wasted copying down the argument and passing it to the function. Also pass by reference gives the function called the permission to read edit the argument directly.

19. How are overloaded functions differentiated by the compiler?

they have to have same return type, but different input parameters

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

Lack of defining a base case, or writing the recursion step incorrectly so that it does not converge on the base case

21. What are the similarities between iteration and recursion?

Incorrectly writing either can result in infinite loops, both perform the same task over and over.

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

statically and dynamically

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

They're basically the same, however, a string ends w/ a null character, denoting the end of the string (and the size). A char array has potential to be any size, so it must be declared or limited.

24. How are arrays passed to functions?

you must pass the array and its size to the function. Ex. function(array, size);

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

Arrays that are static remain the same size throughout the program execution.

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

It depends what parameters you are passing in a multidimensional. A multideminsional can have more than two dimensions.

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

The sorted array or list is built one entry at a time.

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

It finds the minimum element in the array and swaps it with the first element then repeats this step swapping the minimum element with nth positions in the array.

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

Best case-1 Worst Case-N

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

The base case is when the length of the current array is 1.

31. What is a pointer?

a name that represents actual location or value of a data type. int *bPtr points to an integer value.

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

& returns the memory address of its operand.

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

The star operator returns the object at that memory location.

34. How can an array be addressed in pointer/offset notation?

The array can act as a pointer or be referenced by a pointer. *(arrayPtr + 3) or *(array + 3)

35. What does the sizeof operator return?

the size of operand in bytes

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

There are four way to pass pointer to a function 1. non constant pointer to non constant data 2.non constant pointer to constant data. 3. constant pointer to non constant data. 4. constant pointer to constant data.

37. What is a function pointer?

it is a pointer that points to the address of a function.

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

Unlike arrays, linked lists can insert and delete without shifting data and change in size easily.

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

An array is capable of accessing any part of that array based on the index. The link list must be traversed from the beginning or the end, that is, data can only be accessed if it is adjacent to the previous or next nodes.

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

Passing the head pointer of a linked list to a function gives that function access to all nodes of that linked list

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

Basic linked list points to a NULL at the end and circular points to the head at the end.

42. What is the main advantage of a doubly-linked list over a basic linked list?

they have double pointers, so each node can point forwards or backwards. Making it a little easer to get to where you need.

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

The main disadvanate is that if hard to remove all items in the list.

44. What is a stack?

a collection of data to be executed in the order of last in first out

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

push and pop

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

You keep track of the last item entered into the array and do not allow the user access to the other parts of the array.

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

have a linked list, and for each push, add an item onto beginning of the list, and for each pop remove the first element in the list.

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

isEmptry()//determines if the stack is full or not pop()//to remove the top element if it is not the on you are looking for search()//you compare each element of the stack to the data you are looking for.

49. What is a queue?

A queue is an abstract data structure in which operations are performed at both ends giving the queue first in first out behavior.

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

Enqueue and Dequeue

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

you could use the first element of the list as the remove point and insert at the end f you do this you would have to shift the elements down each time you remove an item unless you make the array circular.

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

Implement a linked list that only allows nodes to be added to the tail and can only be retrieved from the head.

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

push

54. What is a tree?

It is a list of numbers in a list made by comparing values of nodes already in the tree and adding to the appropriate spot. Its a list made up of nodes with left and right pointers.

55. What is the height of a tree?

the number of generations or levels the tree has

56. What is a leaf?

a leaf is a node with no children

57. What is a binary search tree?

A binary tree where the search key in any node n is greater than the search key in any node in n's left subtree, but less than the search key in any node in n's right subtree.

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

traverse the left subtree. visit the root. traverse the right subtree.

59. How many comparisons does it take to find an element in a binary search tree?

at the minimum log(n)/log(2), at the maximum N searches, where N is the number of nodes.

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

functions and data members

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

Public, protected, and private.

62. How are objects initialized when they are created?

objects are initialized by stating the classname first then assigning a name to the object. Within the object's ( ) is a place where it can be initialized.

63. What is a function signature?

A function signature is the function's parameters and their type and the name of the function. The return type is not a part of the function signature.

64. What is a recursive function?

A recursive function typically simplifies a problem by calling itself, until arriving at a base case, at which it recursively returns a solution.

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

Any recursive function can be solved using an iterative method.

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

static arrays cannot be changed while normal arrays can be changed

67. 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[]?

An array of strings reads the string literals, meaning anything within "quotes". A char array reads a string as each individual character.

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

The divide-and-conquer technique is where it takes the sorting algorithm and breaks up the array into smaller elements of the array to tackle the problem and run the algorithm.

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

Break a single array down into many arrays with individual elements, then sort the elements as you reconstruct them back into a single array.

70. What is a pointer?

pointer is a programming data type whose value points to another value stored in computer memory by its address.

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

You create and run the algorithm multiple times, while measuring the amount of time it takes, you then average the results.

72. Briefly, how does selection sort work?

goes to the middle of the list, checks to see if it is greater or less then the value given and moves through the list accordingly to add the value into the proper place.

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

Inserts into a linked list are far cheaper than an insert into an array. With an array, all items after the one being inserted must me shifted, with a linked list, only the pointers are replaced.

74. What is a queue?

A queue is a data structure where the first node in is the first node out.

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

Push, Pop, isEmpty, isFull Maybe List to show all elements or an operation to count how many elements in the stack.

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

it runs through the parents and the children in order

77. How do you delete a node from a binary search tree?

if the node has no children, delete it right away, otherwise, put either the furthest right node on the left side or the furthest left node on the right side in that place and perform a the above on that node to guarantee that it's children get handled properly.

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

log(n)