

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

A prototype program is used in problem solving to collect data for the problem.

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

refining, production, maintenance.

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

the main advantages to object-oriented programming is data abstraction, easier maintenance, and re-usability.

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

C++ programs begin execution at the main function.

5. What is a variable?

a variable is an object where data is stored.

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

variables can be declared in classes and methods.

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

A do...while statement will always execute the "do" piece of code at least once before checking the condition. A while statement will always check the condition first.

8. What is typically included in a class definition?

the class name and its public data types such as constructors, methods, functions.

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

Local variables cannot be used outside of that function body. When a function terminates the values of its local variables are lost. Where as data members are variables in a class definition, and they exist throughout the life of the object.

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

A constructor initializes values and a function usual performs some sort of operation.

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

If a class does not explicitly include a constructor, the compiler creates one.

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

as many as wanted/needed

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

A function prototype describes the class's public interface without providing how the function works. A function definition contains the inner workings of the function.

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

A header file is a cross communication method between source files, to help limit the size of each individual program. Without header files a program will be 1 large lump of code and thus fairly hard to debug and traverse.

15. What does a function signature include?

A function signature includes the name of the function and types of arguments, but not the return type.

16. What is the scope of global variables?

Global variable can be accessed by any class with an object in the variables class

17. How does the compiler handle inline functions?

Ignores the inline qualifier and typically does so for all but the smallest functions.

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

The function can change the data stored in a variable passed by reference directly.

19. How are overloaded functions differentiated by the compiler?

overloaded functions must have the same return type but different input parameters

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

Not having a base case that returns a base value.

21. What are the similarities between iteration and recursion?

Both are based on a control statement - Iteration-repetition structure - Recursion-selection structure
Both involve repetition -Iteration-explicitly uses repetition structure -Recursion-repeated function calls
Both involve a termination test -Iteration-loop-termination test -Recursion-base case Both gradually approach termination -Iteration-modifies counter until loop-termination test fails -Recursion-produces progressively simpler versions of problem Both can occur indefinitely -Iteration-if loop-continuation condition never fails -Recursion-if recursion step does not simplify the problem

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

by either inputting the different elements in the array at coding or giving the array specifications as to the size of the array. it is also possible when giving it specifications to tell the array to grow as needed.

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

strings that are used in an char array are much easier to manipulate than as a string object, because each character is stored separately rather than as a whole

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

array it is the collection of similar data types ex: int a[10] ten indicates the size of array. [] is index of array, we can give only integer values to array of a. where as string mean collection of group of characters. string declarations have a datatype usually causes storage to be allocated in memory that is capable of holding some predetermined number of symbols. However Arrays can be declared to contain values of any non reference data type. Multiple arrays of the same type

25. How are arrays passed to functions?

by reference

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

static arrays are available throughout the program

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

All subsequent dimensions after the first one.

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

one by one, each item is taken and placed into the correct place in the sorted portion of the array until there are no more items

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

you loop through the entire list, and for each iteration you find the smallest element and move it to the current position.

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

very simplistic it moves one element from the list by one and inserting them in their correct position into a new slot then starting over. best case is $O(1)$

31. What is a pointer?

Its a location in memory that contains the memory address of another location in memory that contains information.

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

The address (&) operator returns the memory address of its operand.

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

The * returns the value

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

$*(array + element)$

35. What does the sizeof operator return?

sizeof returns the size in bytes of the respective object.

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

Pass by reference with reference arguments. `function(int &)` Pass by reference with pointer arguments. `function(int *)`

37. What is a function pointer?

is a pointer that contains the address of a function

38. What is a linked list?

its a list of pointers that are linked 2 each other. every pointer points to the next pointer which has the next element

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

The size is not fixed, easier sorting because of no shifting, easier to insert items into the list

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

Linked lists are passed as arguments to a function by passing the head pointer of the list by reference.

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

On a circular linked list, the end element points to the head, thus making the circle. A basic link list has nothing pointing to the head, and the last element points to null.

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

Doubly linked lists can be traversed in both directions

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

They use more memory and you have to make 2 pointers and update 2 pointers each time you change the list.

44. What is a stack?

A list that can be added to or removed from only from the top.

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

pop and push

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

Allocate an array of some pre-defined size. Bottom stack element stored at element 0. The last index in the array is the top. Increment top when one element is pushed.

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

use a linked list and keep track of a pointer that points to the first item

48. How are infix expressions evaluated by computers?

convert infix expressions to postfix expression and evaluate the postfix expression

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

A function to keep track of the elements popped and a function to put the popped elements back into the function.

50. What is a queue?

a queue is a storage container that holds its objects in a first in, first out priority

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

Make use of a front pointer and a tail pointer. When the tail pointer reaches the end of an array it gets reset to the front of the array. If the two pointers are equal then the queue is either full or just has one element.

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

By implementing a list in which a tail pointer points to the item most recently inserted onto the back of the list and a head pointer which points to the item least recently inserted onto the list. Also, items can only be removed from the front of the list and can only be added to the back of the list.

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

The stack operation is almost the opposite of the enqueue operation.

54. What is a tree?

an infinite set of nodes that have a root and descendants, and each addition is a leaf.

55. What is the height of a tree?

the number of generations or levels of a tree

56. What is a leaf?

a node with degree 0. last node in the tree and furthestest away from the root.

57. What is a binary tree?

a binary tree is a special search tree where each node has either 2, 1, or 0 children

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

Traverses the binary tree in inorder. It lets you visit the nodes in order according to their data values.

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

$O(\log n)$ in balanced trees

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

public and private variables and initiations of functions included in the class.

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

public and private.

62. How are objects initialized when they are created?

Objects are usually initialized at the beginning of the program and are initialized usually in the main function. They are initialized after the class's name.

63. What is a function signature?

Return type, function name, and the parameter types

64. What is a recursive function?

A function that calls itself. With each function call the problem is diminished.

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

you can also solve the problem iteratively

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

The static array will be available to the whole of the program, whereas an array that is not static is only available within the scope in which it is declared. //pass-by-reference???

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[]?

a string is a variable with a single entity of letters combined into one while in a char[] each letter is assigned a memory value and each has its own variable assigned i.e. char[0] char[1].

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

To divide and conquer, you divide the program in half once or multiple times to the extent that the program as a whole is easier to work in groups to solve the problem.

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

Divide into two sublists and then those sublists break into length 1, then merges the sublists sorted.

70. What is a pointer?

A pointer is a variable that stores the address of another variable.

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

Using some sort of counting principle of the number of operations performed in an algorithm.

72. Briefly, how does selection sort work?

Iterate through the list (assume list of integers), find the smallest one, and put it in a separate "sorted" list... then traverse again and find the next smallest, and move it to the sorted section... and repeat until you run out of elements to sort.

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

a linked list has a dynamic size, but an array only has a fixed size and takes a lot of extra operations to increase its size.

74. What is a queue?

A queue is a collection of elements with the characteristics of first in first out (FIFO). The first item in is the first item out. All additional items are added to the end of the list.

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

Push and Pop

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

The Euler tour traversal of a tree the function that iteratively keeps track of the pointer on each node.

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

You remove the element from the tree and move the next highest element from the left into its place.

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

$\log(n)$