

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

To get early feedback from users in early stages of development. To show users a first idea of what the program will do/look like. To make sure the program will meet requirements before intense programming begins.

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

All stages are influenced except setting the program requirements. If a test fails, it can change the whole design, implementation, etc of a program as well as the final outcome.

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

They make it easier to reuse and adapt previously written code and they separate complex programs into smaller, easier to understand classes.

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

After loading the required include statements and libraries, the main method begins the execution.

5. What is a variable?

A way to store different values into the program, such as numbers, words, letters, etc.

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

anywhere in the code, can be at the top, or in the middle of the code, or anywhere.

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

A while loop terminates at the beginning of the loop, a do/while terminates at the end of the loop.

8. What is typically included in a class definition?

member functions and data members

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

data member is accessible to all functions in the class, where local variable is only available to the member function and value is lost when function closes

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

Constructors don't have a return type.

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

The Compiler creates this automatically for you, unless you specifically create one.

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

It depends on the number of objects. Each object should have a constructor.

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

A function definition is the code that defines the function placed in the {} brackets that determines that function's operation. A function prototype shows the function's public interface without exposing implementation. It shows name, return type and type of parameters.

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

#include "hat.h" allows the use of the hat container, so called from the act of pulling names from a hat. The header file for hat includes functions for storage and retrieval of its contents, mainly put(), peek(), and pull().

15. What does a function signature include?

identification of a function and the data types of its parameters, it has the name, and the data type variables

16. What is the scope of global variables?

They have file scope when placed outside a function.

17. How does the compiler handle inline functions?

When the program compiles, it will copy the function in place, avoiding a function call.

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

Don't have to make copies of stuff.

19. How are overloaded functions differentiated by the compiler?

The compiler selects proper function to execute based on number, types and order of arguments in the function call.

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

Infinite recursion can occur when the base case is omitted or the recursion step is written incorrectly so that it never converges on the base case.

21. What are the similarities between iteration and recursion?

Both involve a controlled repetition structures, and they both have a termination test. Also both of them can loop forever.

22. What are the similarities between iteration and recursion?

both need to have some kind of base case to tell when the loop needs to stop.

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

user defined value or by the number of elements in initializer list

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

Strings declared using the type string can vary in length. Strings declared using an array of characters can not extend past the array length.

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?

not answered

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

one less than the number of dimensions the array contains

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

Insertion sort takes each entity of an unsorted array in order, and sorts it into the correct position between the entities it has sorted.

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

selects the minimum from the remaining elements of the array, and places the minimum in the current position and moves to the next position

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

n operations The best case scenario is when all the numbers are in increasing order.

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

when each half or the original array has nothing else to sort, and puts the halves back together

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

This operator returns the memory address of its operand.

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

It returns a synonym, alias or nickname of the name of the object that its operand points to in memory (dereferencing the pointer).

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

yes `*(array +offset)`

35. What does the sizeof operator return?

sizeof returns the size of the operand in bytes.

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

nonconst -> nonconst const -> nonconst nonconst -> const const -> const

37. What is a function pointer?

Contains the address of the function in memory.

38. What is a linked list?

a series of structures containing data items and a pointer that links the structure to the next structure.

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

its resizeable

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

In arrays you have direct access to each element, but in linked lists you have to traverse the list up to the element that you need.

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

In a circular linked list, every node has a successor which means that the last node's successor is the head or first node.

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

Doubly-linked lists can traverse backwards, though they still cannot access data at any point in the list, it can be shorter to traverse backwards.

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

Each node requires an extra node, requires more memory, and is more difficult to insert and remove individual nodes

44. What is a stack?

a data type that stores a set of elements in a certain order, 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?

for efficiency it is easiest to implement the left to right method, with the right most item being the one that is retrieved when a pop is performed, and a push places an object on the furthest element.

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

adding the element to the end of the list so that it links to the element before it. when popping the element it takes the last element in the list and sets the previous one as the end of the list.

48. How are infix expressions evaluated by computers?

infix expressions are converted to postfix expressions before they can be evaluated by a computer. Operands stay in the same order. Operators always follow the operands.

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

Pop each element off the stack (and into another stack for storage) until the desired element is found.

50. What is a queue?

stores a set of elements where the first element in, is the first element to be removed

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

enqueue and dequeue

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

Boolean isFullQ(queue) ::= rear == MAX_QUEUE_SIZE-1

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

push, which inserts something at the top of the stack.

54. What is a tree?

A connected, undirected graph without cycles.

55. What is the height of a tree?

is the depth of its furthest leaf

56. What is a leaf?

a node with no children.

57. What is a binary tree?

A binary tree can have only two children for each node.

58. What is a binary search tree?

Binary search trees are a fundamental data structure used to construct more abstract data structures such as sets, multisets, and associative arrays.

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

$\log(n)$

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

Function members and member attributes, or in other words, the class variables and functions.

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

Public Private

62. How are objects initialized when they are created?

`classname objectname(input parameters);`

63. What is a function signature?

Includes the name, accepted parameters, and return type.

64. What is a recursive function?

a function that calls itself until it reaches a base case.

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

One can use iteration or by using a simple while/for loop.

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

Static can be initialized and non-static cannot in C++.

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[]?

The programmer must provide the null terminating character in a string of characters using type char[] and must also ensure that the array is large enough to contain the input string of characters and the null terminating character.

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

Divide and conquer means dividing a problem into 2+ problems which are solved and then merged back together for a final solution to the whole problem.

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

It uses the divide and conquer technique recursively and then when merging back together it compares each element together in a sorted list, this is done by reversing the divide and conquer technique

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

merge sort splits an array of elements into smaller and smaller array's till the value of 1 is reached.

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

Try several sets of data on an algorithm that includes worst case, best case, and random cases. Also try and run the same program on a different computer with the same specs

72. Briefly, how does selection sort work?

Selection sort searches the array for the lowest value and swaps it with the first value in the array. Then searches for the next lowest value and swaps it with the second item in the array, and so on.

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

link lists dont have a fixed size

74. What is a queue?

A queue is an abstract data type that involves putting an element at the back of the list. Queues use the FIFO, first in first out, idea where the first element in the list is the first that comes out.

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

push and pop

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

it is where you visit the parent then you visit the children

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

You traverse the tree till you find the node you are wanting to delete. If the node has no children you delete it. If the node has children, before you delete, you find the left-most of its children and attach it to the root then you can delete the node.

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

2^n where n is the # of levels the binary tree has