1. C++ (what is virtual function ? what happens if an error occurs in constructor or destructor. Discussion on error handling, templates, unique features of C++. What is different in C++, (compare with unix).

2. Given a list of numbers (fixed list) Now given any other list, how can you efficiently find out if there is any element in the second list that is an element of the first list (fixed list).

3. Given 3 lines of assembly code : find it is doing. IT was to find absolute value.

4. If you are on a boat and you throw out a suitcase, Will the level of water increase.

5. Print an integer using only putchar. Try doing it without using extra storage.

6. Write C code for (a) deleting an element from a linked list (b) traversing a linked list

7. Compute the number of ones in an unsigned integer.

ANS.

```
#define count_ones(x) \
(x=(0xaaaaaaaa&x)>>1+(0x55555555&x), \
x=(0xcccccccc&x)>>2+(0x33333333&x), \
x=(0xf0f0f0f0&x)>>4+(0x0f0f0f0f&x), \
x=(0xff00ff00&x)>>8+(0x00ff00ff&x), \
x=x>>16+(0x0000ffff&x))
```

8. Compute the discrete log of an unsigned integer.

ANS.

```
#define discrete_log(h) \
(h=(h>>1)|(h>>2), \
h|=(h>>2), \
h|=(h>>4), \
h|=(h>>8), \
h|=(h>>16), \
h=(0xaaaaaaaa&h)>>1+(0x55555555&h), \
h=(0xcccccccc&h)>>2+(0x33333333&h), \
h=(0xf0f0f0f0&h)>>4+(0x0f0f0f0f&h), \
h=(0xff00ff00&h)>>8+(0x00ff00ff&h), \
h=(h>>16)+(0x0000ffff&h))
```

9. Let f(k) = y where k is the y-th number in the increasing sequence of non-negative integers with the same number of ones in its binary representation as y, e.g. f(0) = 1, f(1) = 1, f(2) = 2, f(3) = 1, f(4) = 3, f(5) = 2, f(6) = 3 and so on. Given k >= 0, compute f(k).

10. A character set has 1 and 2 byte characters. One byte characters have 0 as the first bit. You just keep accumulating the characters in a buffer. Suppose at some point the user types a backspace, how can you remove the character efficiently. (Note: You cant store the last character typed because the user can type in arbitrarily many backspaces)

11. Write a function to find the depth of a binary tree.

12. Given two strings S1 and S2. Delete from S2 all those characters which occur in S1 also and finally create a clean S2 with the relevant characters deleted.

13. Assuming that locks are the only reason due to which deadlocks can occur in a system. What would be a foolproof method of avoiding deadlocks in the system.

14. Given an array t[100] which contains numbers between 1..99. Return the duplicated value. Try both O(n) and O(n-square).

15. Given an array of characters. How would you reverse it. ? How would you reverse it without using indexing in the array.

16. An array of integers of size n. Generate a random permutation of the array, given a function rand_n() that returns an integer between 1 and n, both inclusive, with equal probability. What is the expected time of your algorithm?

ANS. "Expected time" should ring a bell. To compute a random permutation, use the standard algorithm of scanning array from n downto 1, swapping i-th element with a uniformly random element <= i-th. To compute a uniformly random integer between 1 and k (k < n), call rand_n() repeatedly until it returns a value in the desired range.

17. An array of pointers to (very long) strings. Find pointers to the (lexicographically) smallest and largest strings.

ANS. Scan array in pairs. Remember largest-so-far and smallest-so-far. Compare the larger of the two strings in the current pair with largest-so-far to update it. And the smaller of the current pair with the smallest-so-far to update it. For a total of <= 3n/2 strcmp() calls. That's also the lower bound.

18. Write an efficient C code for 'tr' program. 'tr' has two command line arguments. They both are strings of same length. tr reads an input file, replaces each character in the first string with the corresponding character in the second string. eg. 'tr abc xyz' replaces all 'a's by 'x's, 'b's by 'y's and so on. ANS.
a) have an array of length 26.
put 'x' in array element corr to 'a'
put 'y' in array element corr to 'b'
put 'z' in array element corr to 'c'
put 'd' in array element corr to 'd'
put 'e' in array element corr to 'e'
and so on.

the code
while (!eof)

```
{
c = getc();
putc(array[c - 'a']);
}
```

19. what is disk interleaving

20. why is disk interleaving adopted