一、单项选择题（本大题共 15 小题，每小题 2 分，共 30 分）

提示：在每小题列出的四个备选项中只有一个是符合题目要求的，请将其代码填写在下表中。错选、多选或未选均无分。

1. Consider the following definition of a recursive function ff.

   int ff(int n)
       { if(n == 0) return 1; return 2 * ff(n - 1); }

   If n > 0, what is returned by ff(n)? (   )

   (A) log2 n   (B) n2   (C) 2n   (D) 2 * n

2. Consider the following C++ code fragment.

   for(i=1, s=0; i<=n; i++)
       { t=1;
          for(j=1; j<=i; j++)
            t=t*j;
          s=s+t; }

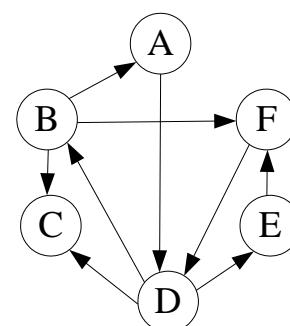   What its asymptotic time complexity? (   )

   (A) O(n)   (B) O(n2)   (C) O(n3)   (D) O(n4)

3. The leaves number of a Huffman Tree is m, the total node number is (   )

   (A) 2m   (B) 2m+1   (C) 2m-1   (D) not certain

4. As shown in the right graph, start from node B, traverse the nodes on a Depth-First Search(DFS) algorithm, which is the possible traverse sequence? (   )

   (A) BADECF   (B) BADCFE

   (C) BACFDE   (D) BADCEF

5. An arithmetic expression (a+b)/c-d*e can be changed to the postfix expression (   )

   (A) abc+/de*-   (B) ab+c/-de*   (C) ab+c/de*-   (D) ab+c/de-*

6. An input into a stack is like 6,5,4,3,2,1. Which output is impossible? (   )

   (A) 5 4 3 6 1 2   (B) 4 5 3 1 2 6   (C) 3 4 6 5 2 1   (D) 2 3 4 1 5 6

7. In linear probing there is a problem that is called (   )

   (A) primary clustering   (B) secondary clustering

   (C) double clustering   (D) none of these

8. Which is not the property of a B-tree of order m? (   )

   (A) The root node has m subtrees at most   (B) All leaf nodes are at the same level

   (C) The keys in every node are ordered   (D) All leaf nodes are connected by links

9. If there exist at least one path between every pair of vertices in a graph, the graph is

known as（　）
(A) Complete graph　　　　　　　　　(B) Disconnected graph
(C) Connected graph　　　　　　　　　(D) Eular graph

10．The best sorting algorithm for picking the top 10 elements in a big data set is（　）
(A) Insertion Sort　　　　(B) Heap sort　　　　(C) Quick sort　　　　(D) Bubble Sort

11．The Linked List is designed for conveniently（　）data item.
(A) getting　　　　(B) locating　　　　(C) finding　　　　(D) inserting

12．Suppose that a linear ordered list contains n=37 nodes, the binary search is applied to the list, the maximum times in searching is（　）
(A) 5　　　　(B) 6　　　　(C) $2^5 - 1$　　　　(D) $2^6 - 1$

13．Insert a node with pointer q before the node with pointer p in the doubly linked list:（　）
(A) p->prev=q; q->next=p; p->prev->next=q; q->prev=q;
(B) p->prev=q; p->prev->next=q; q->next=p; q->prev=p->prev;
(C) q->next=p; q->prev=p->prev; p->prev->next=q; p->prev=q;
(D) q->prev=p->prev; q->next=q; p->prev=q; p->prev=q;

14．On the following data structures,（　）is non-linear data structure.
(A) array　　　　(B) stack　　　　(C) queue　　　　(D) graph

15．In the following sequence,（　）is not a heap?
(A) 100,85,98,77,80,60,82,40,20,10,66　　　(B) 100,98,85,82,80,77,66,60,40,20,10
(C) 10,20,40,60,66,77,80,82,85,98,100　　　(D) 100,85,40,77,80,60,66,98,82,10,20

二、名词解释题（本大题共 4 小题，每小题 4 分，共 16 分）。

提示：解释每小题所给名词的含义，若解释正确则给分，若解释错误则无分，若解释不准确或不全面，则酌情扣分。

1．LIFO

2．heap

3．merge sort

4．BST

## 三、应用题（本大题共 4 小题，1-2 每小题 8 分，3-4 每小题 9 分，共 34 分）

提示：有求解过程的要尽量给出解题步骤，只有最终答案会酌情扣分。

1．Build the Huffman coding tree and determine the codes for the following set of letters and weights:

Letter	A	B	C	D	E	F	G	H	I
Frequency	5	7	9	11	23	27	37	46	52

What is the expected length in bits of a message containing n characters for this frequency distribution?

2. Given values 6, 9, 2, 11, 4, 10, 8, 1, 5, 3, 7, 12, write the first pass steps of Shellsort with increment of 4.

3. Show the result of inserting the value 58 into the following B-tree of order 4.
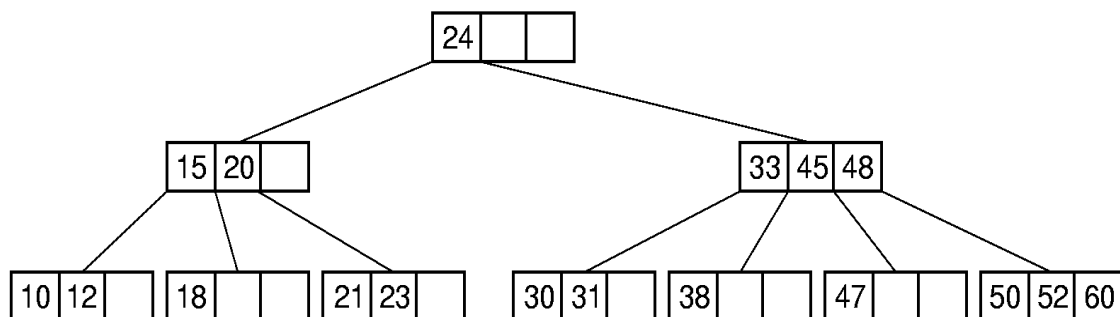
4. Given the following graph, do following:
    (a) Draw the adjacency matrix representation for this graph.
    (b) Draw the adjacency list representation for this graph
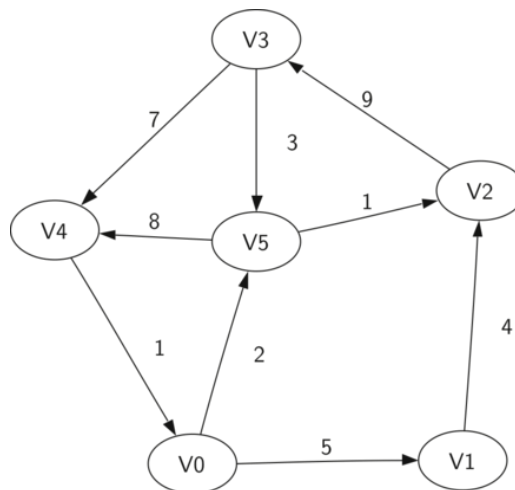
四、编程、设计及分析题（本大题共 2 小题，1 小题 8 分，2 小题 12 分，共 20 分）。

提示：请按照要求写出源程序代码，如果源程序代码中出现语法错误或逻辑错误，则酌情扣分。

1. Judge a binary tree is a BST or not. Use the following node structure.

    typedef struct node{int key; struct node *lchild,*rchild;}bitree;

2. Write a program to calculate an Inverse Poland(逆波兰) expression.