Choose the most appropriate answer. Each correct answer earns 4 points, and each incorrect one deducts 1 points.

1. Which of the following statements is correct about recursive algorithms?
   (A) Recursive algorithms always deal with math formulas.
   (B) Recursive algorithms tend to be shorter and easier to understand.
   (C) Recursive algorithms are easy to design and debug.
   (D) Many recursive algorithms cannot be converted into equivalent iterative ones.
   (E) Compilers can easily optimize recursive algorithms.

2. What is the running time of the following code fragment?

```
for (int i = 0; i < 10; i++)
  for (int j = 0; j < N; j++)
    for (int k = N-2; k < N+2; k++)
      cout << i << " " << j << endl;
```

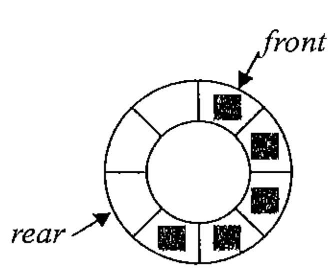
   (A) O(logN) (B)O(NlogN) (C) O(N) (D) $O(N^2)$ (E) $O(N^3)$

3. A linear list is being maintained circularly in an array with *front* and *rear* set up as for circular queues, by the following functions (for adding and deleting an element from the queue).

```
template <class KeyType>
void Queue<KeyType>::Add(const KeyType& x)
// add x to the circular queue
{
   int k = (rear+1) % MaxSize;
   if (front == k) QueueFull();
   else queue[rear=k] = x;
}

template <class KeyType>
KeyType* Queue<KeyType>::Delete(KeyType& x)
// remove front element from queue
{
   if (front == rear) { QueueEmpty(); return 0;}
   x = queue[(++front)%= MaxSize];
   return &x;
}
```

Assume that one such queue was created, and five elements were already added into the queue, as shown in Figure below. How many more elements can be added into the queue before the queue is full?



   (A) 0 (B)1 (C) 2 (D) 3 (E) 4

4. The height of a complete binary tree with n nodes, H_n, can be computed by

   (A) $H_n = n\log_2 n$ (B) $H_n = n\log_2 n + 1$ (C) $H_n = \log_2 n$

   (D) $H_n = \log_2 n + 1$ (E) $H_n = \log_2(n+1) - 1$

參考用

5. Which of the following data structures cannot be used to represent a graph.
(A) Adjacency Lists (B) Adjacency Matrix (C) Incident Matrix
(D) Link List (E)All of the above data structures can be used to represent a graph.

6. What additional requirement is placed on an array, so that binary search may be used to locate an entry?
(A) The array elements must form a heap.
(B) The array must have at least 2 entries.
(C) The array must be sorted.
(D) The size of the array must be a power of two.
(E) The array must contain the searched entry.

7. Suppose we are debugging a quicksort implementation that is supposed to sort an array in ascending order. After the first partition step has been completed, the contents of the array are in the following order:

3 9 1 14 17 24 22 20

Which of the following statements is correct about the partition step?
(A) The pivot could have been either 14 or 17.
(B) The pivot could have been 14, but could not have been 17.
(C) The pivot could have been 17, but could not have been 14.
(D) Both 14 and 17 are pivots in the first partition.
(E) Neither 14 nor 17 could have been the pivot.

8. If we use MergeSort to sort an array with n elements, what is the worst case time required for the sort?
(A) O(1) (B) O(log$n$) (C) O(n) (D) O($n$log$n$) (E)O(n^2)

9. Depth-first search is best implemented using
(A) Recursion (B) A queue (C)A stack (D)A tree (E)A graph

10. If a 5 is inserted to the end of the Min-Heap below, and the heap condition is restored, which position will the 5 go?

Array index i:	0	1	2	3	4	5	6	7	8	9	10
Array content A[i]:	3	9	15	10	17	16	23	19	22	30	33

(A) A[0] (B) A[1] (C) A[2] (D) A[5] (E) A[11]

11. Consider the following recursive function:

```
f(k){
  if (k <= 2) return 1;
  return f(k–1)+f(k–2);
}
```

If the function is called as f(n), where n is a positive integer, then the time complexity of this function is:

(A) $\theta(n)$ (B) $\theta(n^2)$ (C)$\theta(\left(\frac{1+\sqrt{5}}{2}\right)^n)$ (D) $\theta(2^n)$ (E) $\theta(3^n)$.

12. Consider another recursive function:

```
g(p,i,j) {
  if (i == j) return 0;
  m = ∞;
```

注意：背面有試題

```
        for(k=i; k < j; k++) {
        q = g(p,i,k) + g(p,k+1,j) + p[i-1,k,j];

        if (q < m) m = q; }
    return m;
}
```

If the function is called as $g(p,1,n)$ where n is a positive integer and p is a given array of size n, then the time complexity of this function is:

(A) $\theta(n)$ (B) $\theta(n^2)$ (C) $\theta(\left(\frac{1+\sqrt{5}}{2}\right)^n)$ (D) $\theta(2^n)$ (E) $\theta(3^n)$.

13. The following statements about minimum spanning tree (MST) may not be correct. Assume that the weighted graph $G = (V, E)$ is undirected and connected. Do not assume that edge weights are distinct unless this is specifically stated.

S_A: If $|E| \geq |V|-1$, and there is a unique heaviest edge, then this edge cannot be part of any MST of G.

S_B: If G has a cycle with a unique heaviest edge e, then e cannot be part of any MST.

(A) Both S_A and S_B are false. (B) Both S_A and S_B are true.

(C) S_A is true and S_B is false. (D) S_A is false and S_B is true.

14. Consider the following two statements about Huffman encoding scheme:

S_E: If some character occurs with frequency more than 2/5, then there is guaranteed to be a codeword of length 1.

S_F: If all characters occur with frequency less than 1/3, then there is guaranteed to be no codeword of length 1.

(A) Both S_E and S_F are false. (B) Both S_E and S_F are true.

(C) S_E is true and S_F is false. (D) S_E is false and S_F is true.

15. Suppose a file consists of 4 characters occurring with frequencies 31%, 20%, 9%, 40%, respectively. If Huffman encoding is applied to a file consisting of 1000 characters with the given frequencies, what is the length of the encoded file in bits:

(A) 1890 (B) 1900 (C) 1910 (D) 1920 (E) 1930

16. What is the basic concept of the dynamic programming algorithms? (8%) The famous 0/1 knapsack problem is described as follows. Given the capacity m of a knapsack and n objects whose weights are $w_1,\ldots,w_n$ and whose profits are $p_1,\ldots,p_n$, find the largest value of $\sum_{1\leq i\leq n} p_i x_i$ by assigning either 0 or 1 to $x_1,\ldots,x_n$ under the constraint $\sum_{1\leq i\leq n} w_i x_i \leq m$, where $w_1,\ldots,w_n$ and $p_1,\ldots,p_n$, are positive integers. Write a dynamic programming algorithm to solve the 0/1 knapsack problem with the time complexity $O(n\times m)$. You should show that the time complexity of your algorithm is indeed $O(n\times m)$. (12%)

17. One of the fundamental concepts in the NP-completeness theory is the polynomial-time problem reduction. How can we show that a problem X can polynomially reduce to another problem Y? (8%) It is well known that the problem lower bound of the sorting problem is $\Omega(n \log n)$. Please use the problem reduction concept to prove that the problem lower bound of the Euclidean minimum spanning tree problem is also $\Omega(n \log n)$. Note that the Euclidean minimum spanning tree problem is concerned with finding the minimum spanning tree for n planar points. (12%)