

HW2

一. 选择题

1. B;
2. A;
3. D;
4. C;

二. 算法设计题

```
1. 1 void mergeList(Point* head1, Point* head2, Point* merged_head)
2. {
3.     // 预设head1和head2都是升序排列的
4.     // merged_head 要求是降序排列（故采用头插法）
5.     // 注意：链表的头结点都是哑元（不存储实际数据，仅用于简化操作）
6.     Point* p1 = head1->next;
7.     Point* p2 = head2->next;
8.     Point* head = merged_head;
9.     Point* temp = NULL;
10.    head->next = NULL;
11.
12.    while (p1 != NULL && p2 != NULL)
13.    {
14.        if (p1->value < p2->value)
15.        {
16.            temp = p1->next;
17.            p1->next = head->next;
18.            head->next = p1;
19.            p1 = temp;
20.        }
21.        else
22.        {
23.            temp = p2->next;
24.            p2->next = head->next;
25.            head->next = p2;
26.            p2 = temp;
27.        }
28.    }
29.
30.    // 处理剩余节点
31.    Point* p = (p1==NULL) ? p2 : p1;
32.    while (p != NULL)
33.    {
34.        temp = p->next;
35.        p->next = head->next;
36.        head->next = p;
37.        p = temp;
38.    }
39.
40.    // 安全行为
```

```
41     head1->next = NULL;
42     head2->next = NULL;
43 }
```

```
2. 1 void reverseList(Point* head)
2 {
3     // 链表的头结点依旧是哑元
4     Point* past = NULL;
5     Point* current = head->next;
6     Point* future = NULL;
7
8     while (current != NULL)
9     {
10         future = current->next;
11         current->next = past;
12         past = current;
13         current = future;
14     }
15
16     head->next = past;
17 }
```