Exam02 solution.

Fall 2002

1.
(part a)

 (after DecreaseKey)
```
      min
       |
       V
       1                    9     12      10
    /  |   \                |      |
   4   5    7 (T)         15(T)  13(T)
       |   /
       6  8
```
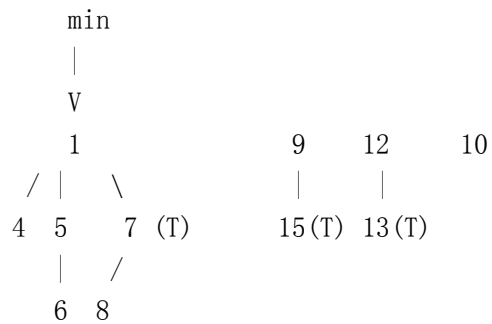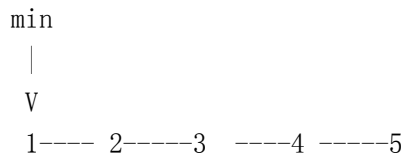
(part b)
   Insert (2^k)+1 elements, where k >= 1, into an empty F-heap.
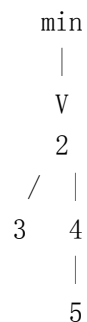After the delete-min operation the F-heap will be one binomial tree, Bk.
Delete every element whose height is greater than 2.
Since we do not perform the cascading cuts, the resulting tree will
have k+1 nodes with degree k

(for example) insert 1,2,3,4,5 into empty F-heap
```
  min
   |
   V
   1---- 2-----3  ----4 -----5
```

  After delete-min operation,
```
            min
             |
             V
             2
           /  |
          3   4
              |
              5
```
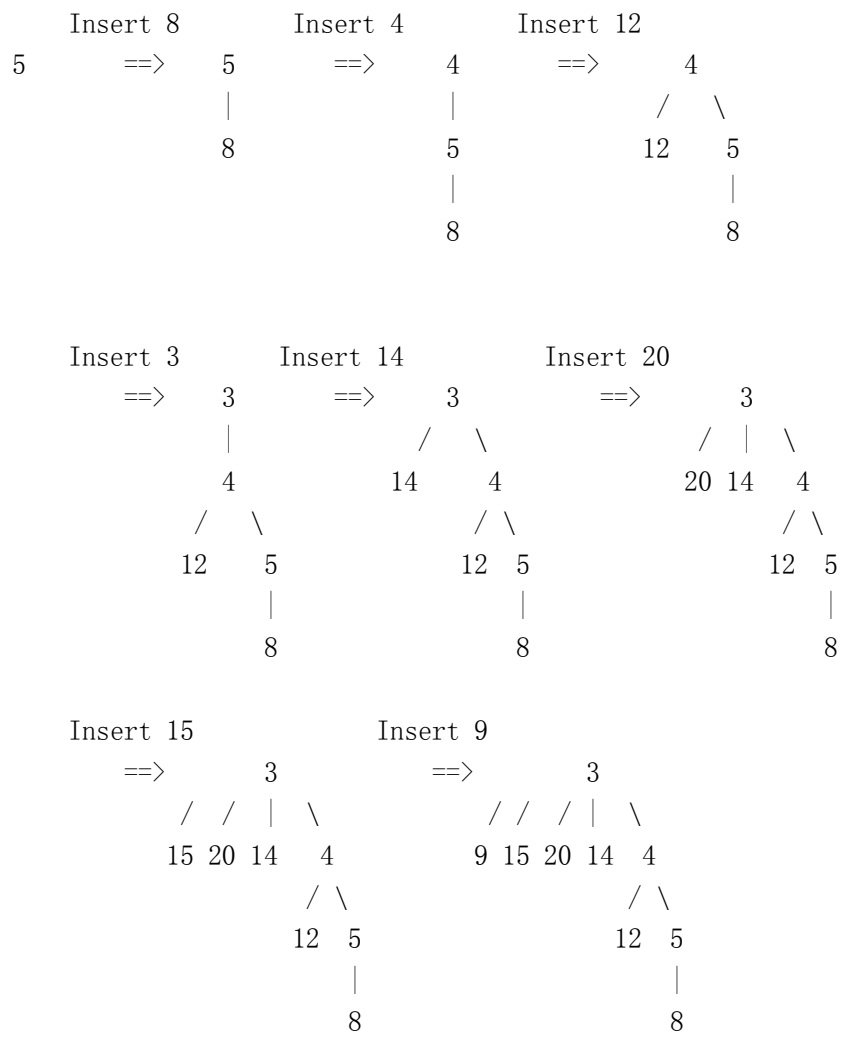
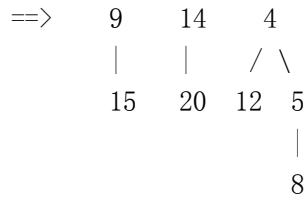   Delete node 5 to get the resulting tree which have k+1 node with degree k

```
        min
         |
         V
         2              degree k = 2   Number of node = K+1 = 3
        / |
      3   4
```

2.

(a)
```
         Insert 8          Insert 4        Insert 12
   5          ==>    5         ==>     4        ==>         4
                     |                 |               /  \
                     8                 5             12    5
                                       |                   |
                                       8                   8
```

```
      Insert 3        Insert 14           Insert 20
        ==>    3        ==>    3              ==>         3
               |              / \                      / | \
               4           14    4                  20 14   4
              / \                / \                      / \
           12    5            12  5                    12   5
                 |                 |                        |
                 8                 8                        8
```

```
      Insert 15              Insert 9
        ==>        3            ==>          3
              / / | \                   / / / | \
           15 20 14   4              9 15 20 14   4
                    / \                        / \
                  12  5                      12   5
                      |                            |
                      8                            8
```
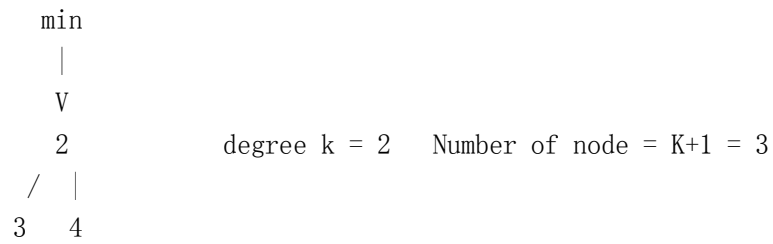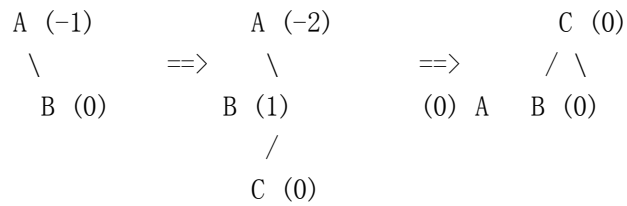
(b) two-pass meld after remove min

```
     9   15   20   14    4
                      / \
                   12   5
                        |
```

```
                      8


==>    9     14     4
       |     |     / \
      15     20   12  5
                      |
                      8


==>             4
            / / | \
           9 14 12  5
           |  |      |
          15  20     8
```
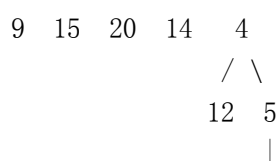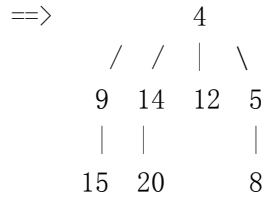
3. RL rotations
   Numbers in parentheses represents balance factors.

   (1) case 1:

```
    A (-1)              A (-2)                 C (0)
     \          ==>      \          ==>        / \
      B (0)               B (1)          (0) A   B (0)
                          /
                         C (0)


    Before              Insert C              After
```

   (2) case 2:

```
        A(-1)                 A(-2)                      B(0):h+2
       / \                   / \                       /       \
    h:Al   C(0)    ==>    h:Al   C(1)    ==>       (0)A          C(-1)
          / \                   / \               /  \   /  \
       (0)B   Cr:h           (-1)B   Cr:h        Al   Bl' Br   Cr
       / \                   / \                 h    h  h-1   h
    h-1:Bl  Br:h-1        h:Bl'  Br:h-1


     Before                Insert to Bl            After
```
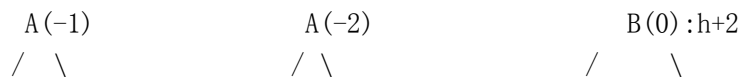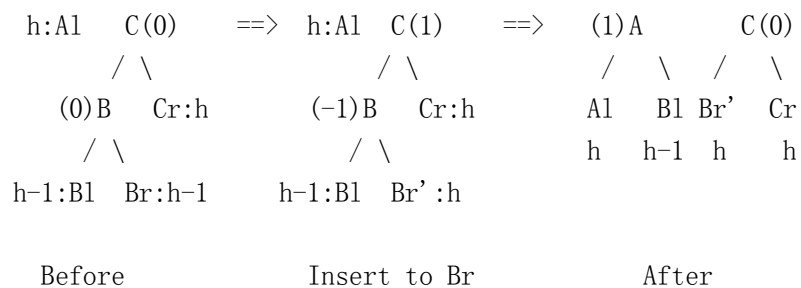
   (3) case 3:

```
        A(-1)                 A(-2)                      B(0):h+2
       / \                   / \                       /       \
```

```
        h:Al   C(0)     ==>  h:Al  C(1)     ==>    (1)A          C(0)
          / \                   / \                / \ / \
      (0)B    Cr:h          (-1)B   Cr:h         Al    Bl Br'  Cr
        / \                   / \                h    h-1  h    h
  h-1:Bl  Br:h-1        h-1:Bl  Br':h


      Before                Insert to Br            After
```
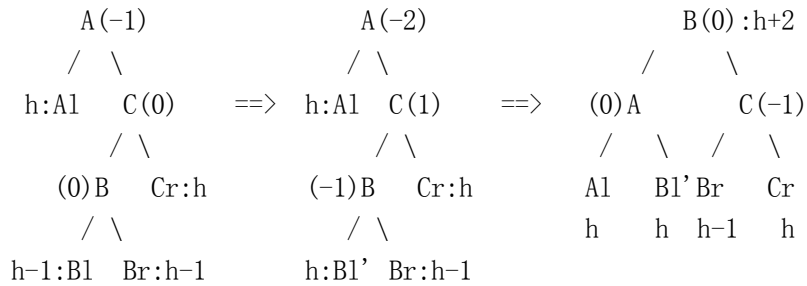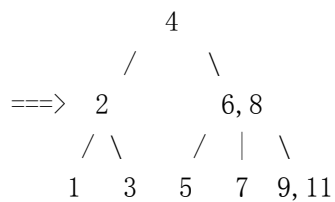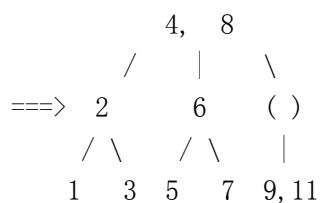
4.
(a) 2-3 tree with 11 keys

```
              4,  8
            /   |   \
         2      6     10
        / \    / \    / \
      1   3  5   7  9   11
```
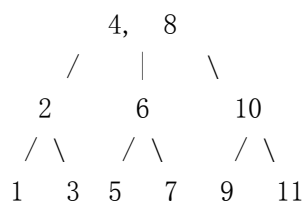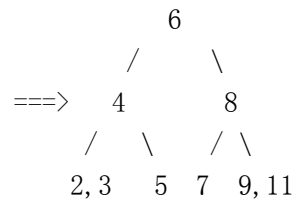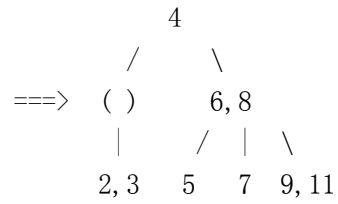
(b) delete "10"
    First, transform deletion from interior to deletion from a leaf.

```
              4,  8
            /   |   \
         2      6      9
        / \    / \    / \
      1   3  5   7  ( )   11
```

```
              4,  8
            /   |   \
  ===>   2      6     ( )
        / \    / \     |
      1   3  5   7  9,11
```

```
             4
           /     \
  ===>   2       6,8
        / \     / | \
      1   3   5   7  9,11
```

(c) delete "1"

```
                4
              /     \
            2         6,8
           / \      /  |  \
         ()   3    5   7  9,11



                4
              /     \
   ===>   ( )        6,8
            |       /  |  \
          2,3      5   7  9,11



                6
              /     \
   ===>    4          8
          / \        / \
        2,3   5     7   9,11
```

5.
Part a:
```
     4                   4                    4
   // \\  insert13     / \    insert 11     / \    insert7
  1    9    ----->   1    9   ------->    1    11 -------->
           RRr            \\    RLb          //\\   LLr
                          13                9   13


     4                    4
   / \\    insert5      / \\
  1    11  ------->    1    11
      / \    LLb           / \
     9   13              7     13
    //                  // \\
    7                   5    9



Part b:

    Rb2              4
 ------->          /   \\
                  1      9
                        / \
                       7    11
```

//
5