ZJU-ADS-HQM2020-WK4

1-1 To implement a binomial queue, the subtrees of a binomial tree are linked in increasing sizes. (1分)

⦿ T   ○ F

1-1 答案错误 ⓘ (0 分)   💡 创建提问

🏅 作者	陈越
单位	浙江大学

1-2 To implement a binomial queue, left-child-next-sibling structure is used to represent each binomial tree. (1分)

⦿ T   ○ F

1-2 答案正确   (1 分)   💡 创建提问

🏅 作者	陈越
单位	浙江大学

ZJU–ADS–HQM2020–WK4

2–1 Which of the following binomial trees can represent a binomial queue of size 42? (1分)

- ○ A. B_0 B_1 B_2 B_3 B_4 B_5
- ⦿ B. B_1 B_3 B_5
- ○ C. B_1 B_5
- ○ D. B_2 B_4

2–1 答案正确   (1 分)   ♀ 创建提问

🏅 作者   陈越
单位   浙江大学

2–2 For a binomial queue, __ takes a constant time on average. (1分)

- ○ A. merging
- ○ B. find–max
- ○ C. delete–min
- ⦿ D. insertion

2–2 答案正确   (1 分)   ♀ 创建提问

🏅 作者   陈越
单位   浙江大学

2–3 Merge the two binomial queues in the following figure. Which one of the following statements must be FALSE? (1分)
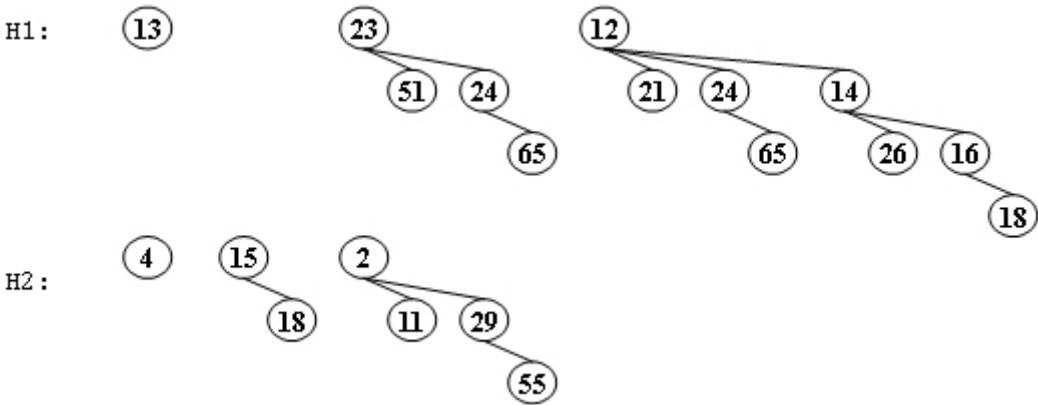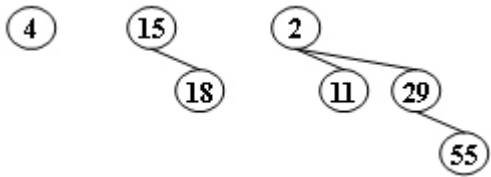

- ○ A. there are two binomial trees after merging, which are B_2 and B_4
- ○ B. 13 and 15 are the children of 4
- ○ C. if 23 is a child of 2, then 12 must be another child of 2
- ⦿ D. if 4 is a child of 2, then 23 must be another child of 2

2–3 答案正确   (1 分)   ♀ 创建提问

🏅 作者   陈越
单位   浙江大学

2–4 Delete the minimum number from the given binomial queues in the following figure. Which one of the following statements must be FALSE? (1分)

- ○ A. there are two binomial trees after deletion, which are B_1 and B_2
- ○ B. 11 and 15 can be the children of 4
- ⦿ C. 29 can never be the root of any resulting binomial tree
- ○ D. if 29 is a child of 4, then 15 must be the root of B_1

2–4 答案正确   (1 分)   ♀ 创建提问

🏅 作者   陈越
单位   浙江大学

ZJU–ADS–HQM2020–WK4

5–1 The functions `BinQueue_Find` and `Recur_Find` are to find `X` in a binomial queue `H`. Return the node pointer if found, otherwise return NULL.

```
BinTree BinQueue_Find( BinQueue H, ElementType X )
{
    BinTree T, result = NULL;
    int i, j;

    for( i=0, j=1; j<=H->CurrentSize; i++, j*=2) {  /* for each tree in H */
        T= H->TheTrees[i];
        if ( X  >= T->Element      (2分) ){  /* if need to search inside this tree */
            result = Recur_Find(T, X);
            if ( result != NULL ) return result;
        }
    }
    return result;
}

BinTree Recur_Find( BinTree T, ElementType X )
{
    BinTree result = NULL;
    if ( X==T->Element ) return T;
    if ( T->LeftChild!=NULL ){
        result = Recur_Find(T->LeftChild, X);
        if ( result!=NULL ) return result;
    }
    if ( T->NextSibling != NULL    (2分) )
        result = Recur_Find(T->NextSibling, X);
    return result;
}
```

5–1 答案正确   (4 分)   💡 创建提问

🏅 作者	**陈越**
单位	浙江大学
时间限制	400 ms
内存限制	64 MB