Karadeniz Technical University
Department of Computer Engineering
Lecturer Omer CAKIR

COM 205 Data Structures
Final Exam, 01.10.2013, 10:00, D-1, D-8
Time : **100** Minutes

```cpp
void print1(DoublyNode* node)
{
   cout << node->elem << endl;
   if (node->next == trailer) return;
   else print1(node->next);
}

void print2(DoublyNode* node)
{
   if (node == trailer) return;
   else print2(node->next);
   cout << node->elem << endl;
}

void main()
{
   DoublyLinkedList list;
   list.insertOrdered("Paul", 720); // küçükten
   list.insertOrdered("Rose", 590); // büyüğe
   list.insertOrdered("Anna", 660); // sıralı ekle

   list.print1(list.header->next);       //(1)
   list.print2(list.header->next);       //(2)
   list.print3(list.trailer->prev);      //(3)
   list.print4(list.trailer->prev);      //(4)
}
```

```cpp
void removeOrdered(const string& e, const int& i)
{
  DoublyNode* current = header->next;

  while (current != trailer)
  {
   if((current->elem == e) && (current->score == i))
   {
    ...........................................;

    ...........................................;

    delete current;
    return;
   }
   current = current->next;
  }

  cout << e << " is not found" << endl;
}
```

2. Complete **removeOrdered()** function above that removes an element from a doubly linked list. **(10P)**

1. **a)** Write down the outputs of the functions **print1()** and **print2()** when called with **header->next** parameter in the **main()** function. **(10P)**
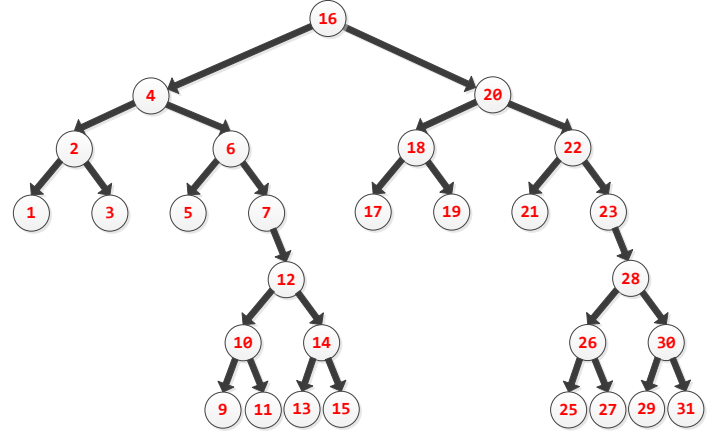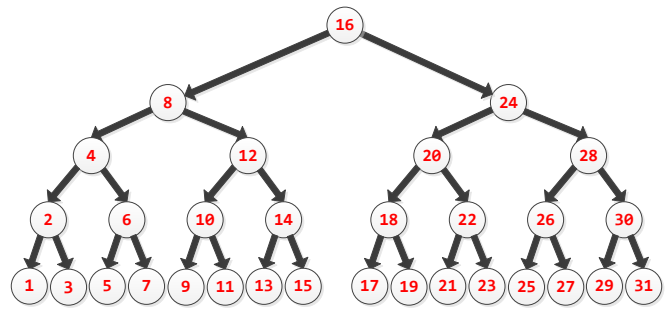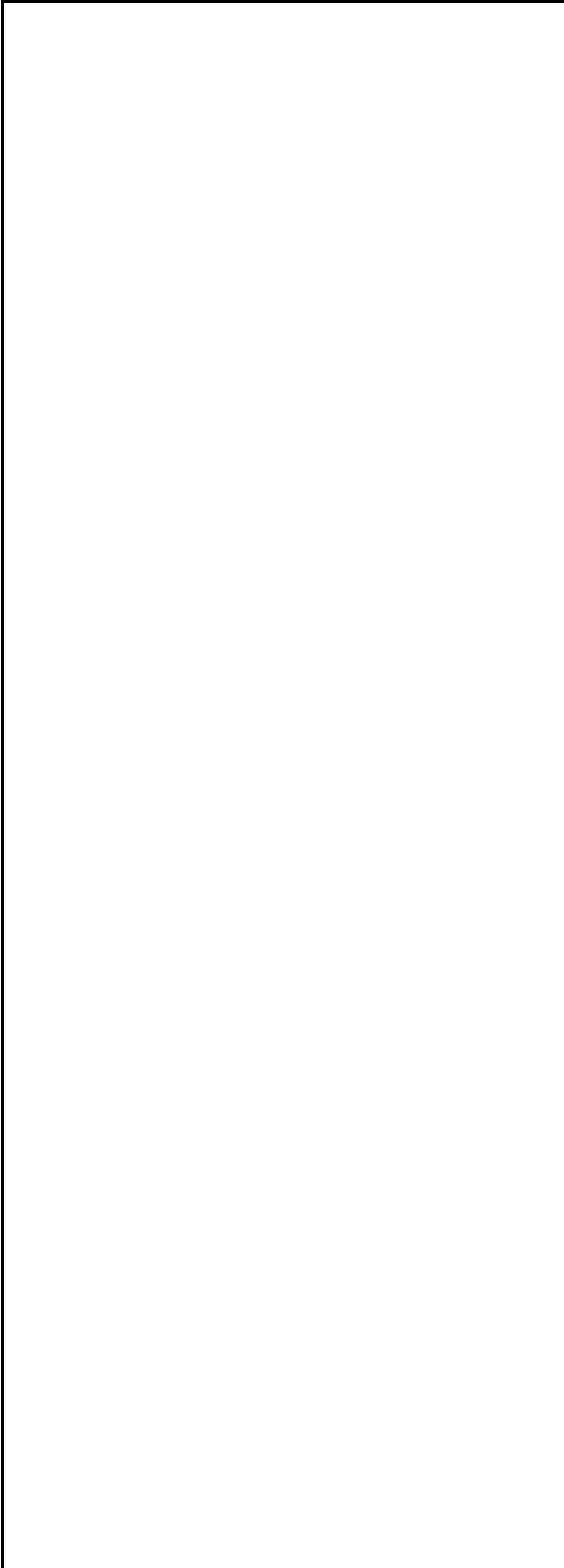
print1()	Print2()

b) Write down **print3()** that is equavalent to **print1()** and **print4()** that is equavalent to **print2()** when called both with **trailer->prev** parameter in the **main()** function.
Hint → Answers take **3** lines.

```cpp
void print3(DoublyNode* node)  // print1()
{
                                              //5P
                                              //5P
                                              //5P
}
void print4(DoublyNode* node)  // print2()
{
                                              //5P
                                              //5P
                                              //5P
}
```

6 5 4 3 2 1 7

3. Insert the elements above into a splay tree? **(20P)**

Tree 1:
16
8, 24
4, 12, 20, 28
2, 6, 10, 14, 18, 22, 26, 30
1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31

Tree 2:
16
4, 20
2, 6, 18, 22
1, 3, 5, 7, 17, 19, 21, 23
12, 28
10, 14, 26, 30
9, 11, 13, 15, 25, 27, 29, 31

4. Complete codes below that deletes 8 and 24 from binary tree like above. **(30P)**

```
if( p->left != NULL && p->right != NULL)
{
    if(parent->left == p)
    {
        parent->........ = p->........;
        p->............. = parent;
        temp            = p->........;
        while(temp->...... != NULL) temp = temp->......;
        temp->.......... = p->........;
        temp->.......... = temp;
    }
    else
    {
        parent->........ = p->........;
        p->............. = parent;
        temp            = p->........;
        while(temp->...... != NULL) temp = temp->......;
        temp->.......... = p->........;
        temp->.......... = temp;
    }

    delete p;
}
```