



NUMBER : .....	NAME : .....	EVALUATION	
		[.....]	.....
SIGNATURE : .....			
Exam Execution Instructions of Faculty of Engineering should be obeyed. Questions are related to 1,4,12 of <a href="#">Program Learning Outcomes</a>			

```
void insertOrdered(SinglyNode* newNode,
                  SinglyNode* current)
{
    if(.....)
    {
        newNode->next = current->next;
        current->next = newNode;
    }
    else
        insertOrdered(newNode, current->next);
}

int main()
{
    SinglyLinkedList list; SinglyNode* newNode;

    list.head = new SinglyNode;
    list.head->score = 0;
    list.head->next = NULL;

    newNode = new SinglyNode;
    newNode->elem = "Paul"; newNode->score = 720;
    list.insertOrdered(newNode, list.head);

    newNode = new SinglyNode;
    newNode->elem = "Rose"; newNode->score = 590;
    list.insertOrdered(newNode, list.head);

    newNode = new SinglyNode;
    newNode->elem = "Anna"; newNode->score = 660;
    list.insertOrdered(newNode, list.head);

    newNode = new SinglyNode;
    newNode->elem = "Mike"; newNode->score = 1105;
    list.insertOrdered(newNode, list.head);
}
```

8 4 12 2 6 10 14 1 3 5 7 9 11 13 15

2. Assume that the numbers above are inserted into a binary tree. Assuming again that another 3 new binary trees are generated by the output of the inorder, preorder and postorder traversals of this binary tree, which of these 3 new trees is exactly the same as the first binary tree? (25P)

*Yanlış cevaptan 5P kılacaktır.*

- (A) inorder
- (B) preorder
- (C) postorder

1. Complete the function `insertOrdered()`. (25P)

*You'll loose 5P from wrong answer.*

- (A) `if ((current == NULL) || (newNode->score <= current->score))`
- (B) `if ((current->next == NULL) || (newNode->score <= current->score))`
- (C) `if ((current == NULL) || (newNode->score <= current->next->score))`
- (D) `if ((current->next == NULL) || (newNode->score <= current->next->score))`

```

void insertOrdered(const string& e, const int& i)
{
    DoublyNode* newNode      = new DoublyNode;
    newNode->elem             = e;
    newNode->score            = i;

    DoublyNode* current = header->next;

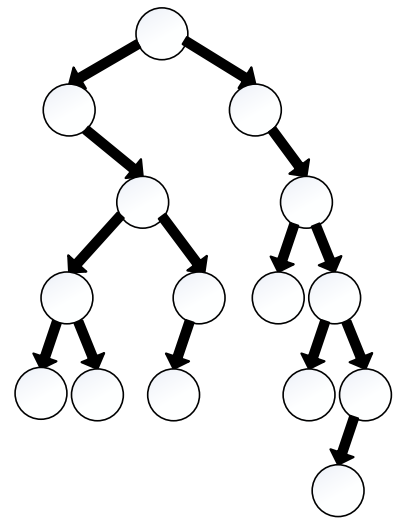
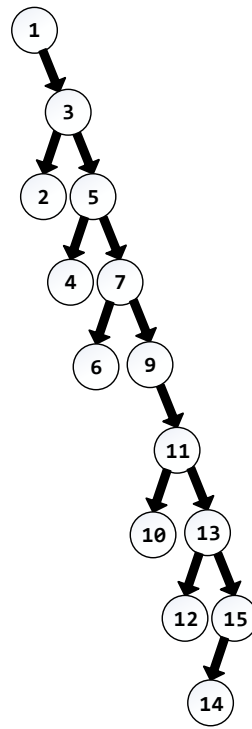
    while (current != trailer)
    {
        if (newNode->score >= current->score)
            current = current->next;
        else
            break;
    }

    newNode->next      = current;
    newNode->prev      = current->prev;
    .....            = .....;
    .....            = .....;
}

```

3. Considering the two lines of the `insertOrdered()` function that are indicated by "...", which of the following choices add a node to a doubly linked list erroneously? (25P) *You'll loose 5P from wrong answer.*

- (A) `current->prev->next = newNode;`  
`newNode->next->prev = newNode;`
- (B) `current->prev->next = newNode;`  
`current->prev = newNode;`
- (C) `current->prev = newNode;`  
`current->prev->next = newNode;`
- (D) `newNode->prev->next = newNode;`  
`newNode->next->prev = newNode;`
- (E) `newNode->next->prev = newNode;`  
`newNode->prev->next = newNode;`



4. Insert 8 into a Splay tree above left. Write the final splay tree above right. (25P)