

2019학년도 제 1학기 기말시험 [좌석번호:]										감독교수 확인란			
시험과목		강좌번호		400650-003		교과목명		데이터구조		담당교수		이광의 (인)	
응시대상		학과		응용SW공학		학년		2		학번		성명	

1. 다음은 배열을 기반으로 하는 양의 정수를 위한 스택이다. 코드를 완성하시오. 단, 오류 시 "0"리턴

```

class stack {
public:
    stack (int s=100) {
        size=s; storage=new int[s]; top=0; }
    bool Push (int el);
    int Pop ( );
private:
    int * storage;
    int size;
    int top
};

```

a. 함수를 완성하시오. [4+4]

```

bool stack::Push (int el) {

};

int stack::Pop ( ) {

};

```

b. DropDownStack은 스택이 꽉 찬 경우 오류를 리턴하지 않고 가장 오래된 원소를 삭제하고 새로운 원소를 삽입한다. 이를 위한 Push 함수를 수정하시오.[5]

```

bool Stack::Push (int el) {

};

```

2. 다음은 10개의 칸을 갖는, 양의 정수 큐 클래스의 일부분이다. 코드에서 변수는 보여주지 않으나 적절히 선언되어 있음을 가정한다.

```

class queue10 { // 삭제 시 앞으로 이동하는 큐~~
public:
    queue10 ();
    bool IsFull();
    bool IsEmpty();
    bool InsertQ (int el); // 성공 true, 실패 false
    int DeleteQ (); // 오류 시 0 리턴
};

```

위의 queue10 20개 활용하여 queue200 클래스를 다음과 같이 구현하고자 한다.

a. 함수를 완성하시오. [14=3+2+2+3+3]

```

class queue200 {
public:
    queue200 ();
    bool IsFull();
    bool IsEmpty();
    bool InsertQ(int el); // 성공 true, 실패 false
    int DeleteQ(); // 오류 시 0 리턴
private:
    queue10 * q[20];
};

```

b. 다음의 값들을 순서대로 삽입한 후의 이진탐색트리를 그리시오. (5, 8, 7, 9, 6, 2, 1, 4, 3, 10) [3]

```
class node {
public:
    node (int data, node * l=0, node * r=0) {
        d = data; lc = l; rc = r;
    }
    int d;                // data
    node *lc, *rc;        // left child, right child
};

class bst {
public:
    bst () { root = 0; };
    int Diff (); // (최댓값 - 최솟값) 계산
    void Print () { Print (root); };
    void Print (node * ptr); // 감소하는 순으로 출력
private:
    node * root;
};
```

a. 함수 Diff()와 Print(node * ptr)을 구현하시오. [6=3+3]

c. 위의 결과로 주어진 트리에서 합병에 의한 삭제 (DeleteByMerging)를 적용하여 4, 8, 5를 순서대로 삭제한 후의 트리(3개)를 그리시오. [4]

4. 다음은 배열에 저장된 **최소힙**이다. 단, “-”는 데이터가 없음을 나타낸다. 초기상태에서 각각 Insert(3), Insert(2), Delete(), Delete()의 연산이 순서대로 실행된 후의 힙의 상태를 기술하시오.[8=3+1+3+1]

[illegible]