

## 자료구조 기말고사

2018-1

1. 스택의 응용에서 다음의 수식을 후위 표기법으로 표기했을 때 옳은 것은  
 $((A/B)+C)-(D * E)$

가.  $A/B+C-D * E$    나.  $AB/C+DE*-$    다.  $A/B+C-*DE$    라.  $AB/C+-DE*$

2. 다음은 스택에 자료를 삽입하는 알고리즘이다. 빈칸에 적합한 내용은?

```
procedure insert(data, n, top, stack)
    if top >= n then call stackFull;
    top = top + 1;
    stack(top) = (      );
```

가. top   나. data   다. top-1   라. data-1

3. 원형 큐의 삭제 알고리즘을 순서대로 늘어놓아라. (예 3-2-1, 1-3-2 등 )

- 1) item=Q[front];
- 2) front=(front+1)mod n;
- 3) if(front==rear) then Q empty()

4. 크기가 5인 선형 큐에서 다음의 연산을 수행한다. 큐가 포화상태가 되는 시점을 설명하라.

A삽입-B삽입-C삽입-삭제-삭제-D삽입-삭제-E삽입-삭제-삭제-F삽입

5. 텍에 대한 설명으로 옳은 것들을 모두 골라라.

- 1) 양끝에서 노드의 삽입과 삭제가 가능하다.
- 2) 하나의 포인터를 사용한다.
- 3) Delete ended queue의 약자이다.
- 4) 선형자료구조이다.

6. 다음 중 스택에 대한 올바른 설명을 모두 골라라.

- 1) 스택은 FIFO(First In First-Out)방식으로 동작한다.
- 2) 스택은 양쪽 끝을 사용하여 입출력을 한다.
- 3) 함수 호출시 복귀 주소를 저장하는데 사용된다.
- 4) 배열을 사용하여 구현할 수 있다.

7. 배열로 구현된 스택에 항목들을 삽입하고 삭제하는 연산은 시간복잡도가 어떻게 되는가.

- 1)  $O(1)$    2)  $O(\log n)$    3)  $O(n)$    4)  $O(n^2)$

8. 순서가 A, B, C, D로 정해진 입력 자료를 스택에 입력하였다가 출력할 때, 가능한 출력 순서의 결과가 아닌 것은?

- 1) DABC   2) ABCD   3) ABDC   4) BCDA

9. 스택의 응용 분야로 거리가 먼 것은?

- 1) 미로 찾기    2) 수식 계산 및 수식 표기법    3) 운영체제의 작업 스케줄링    4) 재귀

10. 다음 알고리즘을 수행한 결과를 적어라. 이 알고리즘의 역할은 무엇일까?

Algorithm : unknown

```
main(){
    stack(S) : S는 스택을 의미함
    number=11
    while(number!=0){
        remainder=number % 2
        push(S,remainder) //스택 S에 remainder를 push
        number=number/2
    }
    while(not empty(S)){
        pop(S,x) //스택 S에서 x를 pop
        print(x) // x를 인쇄
    }
}
```

11. 다음의 후위표기식의 결과를 적어라.

10 20 5 + 5 \* -

12. 원형큐에서 front가 5이고 rear=3라고 하면 현재 원형큐에 저장된 요소들의 개수는?(단 원형큐크기는 8)

13. 컴퓨터나 핸드폰의 화면을 위로 올리거나, 아래로 내리는 스크롤(scroll) 동작은 어떤 자료구조의 기능을 하는가?

- 1) 스택    2) 큐    3) 데크    4) 탐색트리    5) 답 없음

14. 연결리스트의 push 연산을 연결리스트로 작성하였다. 빈 칸을 채워라.

```
push(stack, item)
newNode ←getNode();
newNode.data ←item;
(    채워라    )
top ←newNode
end push()
```

15. 연결리스트의 pop 연산을 연결리스트로 작성하였다. 빈 칸을 채워라.

```
pop()
if (stackTop = null) then return null
else {
```

```

        item ← stackTop.data
        (   채워라   )
        return item;
    }
end pop()

```

16. 원형 큐의 enqueue()를 배열로 구현하였다. 빈 칸을 채워라  
 enqueue(q, item)  
 ( 빈 칸을 채워라. 세줄정도가 될 것이다)

```

end enqueue()

```

17. 원형 큐의 dequeue()를 배열로 구현하였다. 빈 칸을 채워라  
 dequeue(q)  
 ( 빈 칸을 채워라. )

```

end dequeue()

```

18. 테크에서 다음과 같은 작업을 수행하고 난 결과를 적어라.  
 InsertFirst(5) - InsertFirst(10) - InsertLast(20) - InsertLast(DeleteFirst())

19. 연결리스트를 사용하여 큐의 enqueue()를 구현하였다. 빈칸을 채워라  
 enqueue(q, item)  
 newNode ← getNode();  
 newNode.data ← item;  
 newNode.link ← null;  
 if (rear = null) then {  
 (            )  
 (            )  
 }  
 else {  
 (            )  
 (            )  
 }  
end enqueue()

20. 연결리스트를 사용하여 큐의 dequeue()를 구현하였다. 빈 칸을 채워라.  
 dequeue(q)  
 if (front = null) then queueEmpty();  
 else  
 (            )  
 (            )  
 if (front = null) then rear ← null;  
 return item;  
 }  
end dequeue()