화면기능정보공학전공 2019107783 김성환
2학년

2019학년도 중간고사 자료구조

1. Boolean의 추상 자료형의 객체 및 OR 연산자를 정의하시오.(5점)
   1) 객체 정의: 참(true) 또는 거짓(false)을 담고 있는 것
   3) OR(b1, b2) := [불을자료형 b1과 b2가 모두 거짓이라면 거짓, 아니라면 참을 반환한다. b1과 b2의 불을자료형을 비교해 b1 또는 b2가 참이라면 참이고, 아니면 거짓이다.]

2. 다음의 빅오 표기법들을 실행 시간이 적게 걸리는 것부터 많이 걸리는 순서로 나열하시오.(5점)
   $O(n^2\log n)$, $O(n^5)$, $O(2^n)$, $O(n!)$, $O(\log n + 100)$
   $O(\log n + 100) < O(n^2\log n) < O(2^n) < O(n^5) < O(n!)$

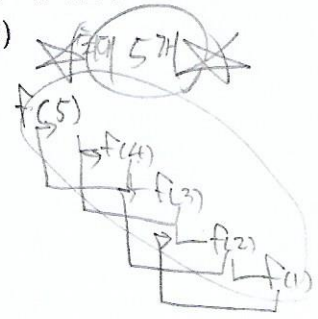
3. 다음 프로그램에서 단위연산을 지정하고 시간 복잡도를 빅오 표기법으로 나타내시오.(10점)
   (1)
   ```
   for (i=0; i<n; ++i)
       for (j=0; j<n; j++)
           for (k=0; k<100; k++)
               sum += i*j*k;
   ```
   단위연산: $99n(n-1) = 99(n^2-n) = 99n^2-99n$
   시간복잡도: $O(n^2)$

   (2)
   ```
   void sum(int n)
   {
       int i,b;
       b=4;
       i=1;
       while(i <= n){
           i = i*b;
       }
   }
   ```
   단위연산: $n/4$
   시간복잡도: $O(n)$

4. 배열에 정수가 들어 있다고 가정할 때 다음 작업의 최악, 최선의 시간 복잡도를 빅오 표기법으로 표시 하시오.(10점)
   (1) 배열의 n번째 숫자를 화면에 출력한다. 최악: $O(1)$ 최선: $O(1)$
   (2) 배열의 모든 숫자를 더한다. 최악: $O(n)$ 최선: $O(n)$
   (3) 배열에서 특정 수 k를 찾는다. 최악: $O(n)$ 최선: $O(1)$ → 순차탐색(linear search) 기준.
   (4) 배열에 원소를 삽입한다. 최악: $O(n)$ 최선: $O(1)$
   (5) 배열에 원소를 삭제한다. 최악: $O(n)$ 최선: $O(1)$

5. 팩토리얼을 계산하는 순환호출 함수 factorial에서 매개변수로 5를 주었다면 최대 몇 개의 factorial 함수의 활성 레코드가 동시에 존재할 수 있는가?(5점)

```
int factorial(int n)
{
    if( n <= 1 ) return(1);
    else return (n * factorial(n-1) );
}
```



6. 다음을 계산하는 순환 및 반복 프로그램을 작성하시오.(10점)

$1 + 2 + 3 + \ldots + n$

$$sum(n) = \begin{cases} 1 & \text{if } n = 1 \\ n + sum(n-1) & otherwise \end{cases}$$

1) 순환 프로그램

```
int sum(int n)
{
    if (n==0) return 0;
    else if (n==1) return 1;
    else return (n+sum(n-1));
}
```

2) 반복 프로그램

```
int sum(int n)
{
    int i, sum= 0;
    for(i=1; i<=n; i++) {
        sum += i;
    }
    return sum;
}
```

7. 다음 물음에 답하시오.(10점)

1) int i=10; int *p; p=&i; *p--;의 문장이 수행되면 i값은 얼마인가? (2)

(1) 11 (2) 10 (3) 9 (4) 8

2) int a[3]; int *p; p=a+1; *++p=5; 의 문장이 수행되면 변경되는 배열의 요소는? (3)

(1) a[0] (2) a[1] (3) a[2] (4) 없음

3) int a[3]; int *p; p=a+1; *p++; 의 문장이 수행되면 변경되는 배열의 요소는? (4)

(1) a[0] (2) a[1] (3) a[2] (4) 없음

4) int a[3]; int *p; p=a+1; ++*p; 의 문장이 수행되면 변경되는 배열의 요소는? (2)

(1) a[0] (2) a[1] (3) a[2] (4) 없음

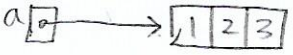
5) int a[3]; int *p; p=a+1; ++*(p+1); 의 문장이 수행되면 변경되는 배열의 요소는? (3)

(1) a[0] (2) a[1] (3) a[2] (4) 없음

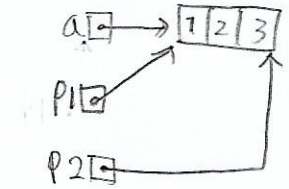
8. 다음의 각각의 문장들을 수행한 다음에 배열과 포인터들이 서로 연결된 모습을 단계별로 그리시오.(5점)

```
void main() {
        int a[]= {1, 2, 3};    //1단계
        int *p1=a, *p2=a+2;//2단계
        *p1 = *p2;             //3단계
        p2 = p1;               //4단계
}
```

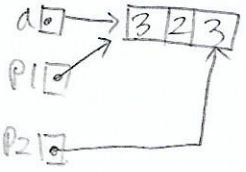
<1단계>



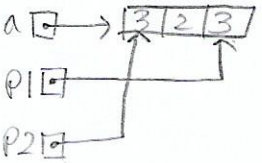
<2단계>



<3단계>



<4단계>



9. 다음 문장들 중 **맞는** 번호를 모두 골라 쓰시오.(10점) 3, 5, 10

(1) 스택에서 삽입 작업이 발생하면 top의 값은 1 감소한다. X

(2) 10, 20, 30, 40, 50 을 스택에 넣었다가 3개의 항목을 삭제하였다. 남아 있는 항목은 40, 50 이다. X

(3) 스택은 후입선출(LIFO) 방식으로 동작한다. O

(4) 스택은 중간에 요소를 삽입하는 것을 허용한다. X

(5) 스택을 초기화하는 연산의 시간 복잡도는 O(1)이다. O
↳ top을 그냥 -1로 변경!?

(6) 스택에 항목들을 삽입하고 삭제하는 연산의 시간 복잡도는 O(n)이다. X

(7) 배열로 구현된 스택에서 top이 0이면 스택에 저장된 요소들의 개수는 0이다 X
↳ 공백상태이면 top은 -1임. O(1)?
stack[0]에 저장된 상태, 1개 존재하는 상황임.

(8) 크기가 MAX_STACK_SIZE인 배열로 구현된 스택에서 포화상태에 해당하는 조건은 top == MAX_STACK_SIZE 이다. X
↳ (MAX_STACK_SIZE - 1)일 때 포화상태이다!! (top이 끝위치에서 배열의 index니까)

(9) 회사에서 입사순으로 승진시킬 때 스택을 사용한다. X
↳ FIFO로 해야지 예를...

(10) 스택은 한쪽 끝을 사용하여 입출력을 한다. O

10. 구조체 배열 지역 변수로 구현된 스택에서 다음의 함수를 완성하시오. 공백 스택일 경우 top= -1이다.(10점)

```
#define MAX_STACK_SIZE 100
typedef int element;
typedef struct {
    element data[MAX_STACK_SIZE];
    int top;
} StackType;

void push(StackType *s, element item)
{
    if (is_full(s)) {
        fprintf(stderr, "스택 포화 에러\n");
        return;
    }
    //아래를 완성하시오.
    else s->data[++(s->top)] = item;

}
// 삭제함수
element pop(StackType *s)
{
    if (is_empty(s)) {
        fprintf(stderr, "스택 공백 에러\n");
        exit(1);
    }
    //아래를 완성하시오.
    else return s->data[(s->top)--];

}
element peek(StackType *s)
{
    if (is_empty(s)) {
        fprintf(stderr, "스택 공백 에러\n");
        exit(1);
    }
    //아래를 완성하시오.
    else return s->data[(s->top)];

}
```

11. 후위표기식 abc*a/de-+*를 계산하는 프로그램에서 스택의 내용을 단계적으로 그리시오. a=2, b=3, c=2, d=5, e=4 이다.(5점)

토큰	스택						
	[0]	[1]	[2]	[3]	[4]	[5]	[6]
a=2	2						
b=3	2	3					
c=2	2	3	2				
*	2	6					
a=2	2	6	2				
/	2	3					
d=5	2	3	5				
e=4	2	3	5	4			
−	2	3	1				
+	2	4					
*	8						

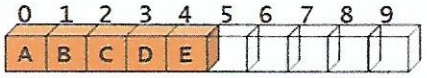
a=2
b=3
c=2
d=5
e=4

12. 다음의 중위 표기식을 후위 표기식으로 변환시 스택의 내용을 단계적으로 그리시오.(5점)
a/b+(c-d)*e

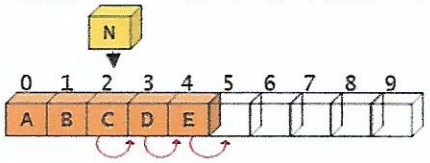
토큰	스택						
	[0]	[1]	[2]	[3]	[4]	[5]	출력
a							a
/	/						a
b	/						ab
+	+						ab/
(	+	(					ab/
c	+	(					ab/c
−	+	(	−				ab/c
d	+	(	−				ab/cd
)	+						ab/cd-
*	+	*					ab/cd-
e	+	*					ab/cd-e
							ab/cd-e*+

13. 다음 배열에서 삽입 및 삭제 함수를 완성하시오.(10점)

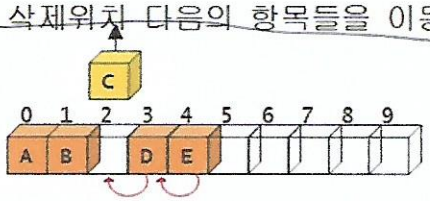
- 1차원 배열에 항목들을 순서대로 저장
  - L=(A, B, C, D, E)



- 삽입연산: 삽입위치 다음의 항목들을 이동하여야 함.



- 삭제연산: 삭제위치 다음의 항목들을 이동하여야 함




```
#define MAX_SIZE 10
int two[MAX_SIZE], size;
void insert(int idx, int value)
{
    if (size >= MAX_SIZE) { // size는 배열 two[]에서 유효한 요소의 개수이다.
        printf("삽입할 수 없습니다\n");
        return;
    }
    //아래를 완성하시오
    else{
        for(int i = (size -1); i >= idx; i--)
            two[i+1] = two[i];
        two[idx] = value;
        size++;
        }
}
int del(int idx)
{
    if (idx<0 || idx >= size) {
            printf("삭제할 수 없습니다\n");
            return -1;
    }
    //아래를 완성하시오
    int tmp;
    tmp=two[idx];
    for (int i = idx; i <(size-1); i++)
        two[i] = two[i+1];
    size--;

    return tmp;
}
```

└ 지워낸 걸 반환하는 건가...?

문제조건에 따로 뭐라 말이 없어서 ...

2019학년도 기말고사 자료구조

1. 다음 문장들 중 맞는 번호를 모두 고르시오.(10점) 1, 2, 5, 9

(1) 10, 20, 30, 40, 50을 큐에 넣었다고 가정하고 3개의 항목을 삭제하였다. 남아 있는 항목은 40, 50 이다. O

(2) 큐는 FIFO 방식으로 동작한다. O

(3) 큐의 삭제연산 보다 스택의 삭제 연산이 훨씬 쉽다. X

(4) 큐는 연결 리스트로 구현할 수 없다. X

(5) 크기가 8인 원형 큐에서 front가 7이고 rear가 2라고 하면 현재 원형큐에 저장된 요소들의 개수는 2 이다. O

(6) 원형 큐에서의 포화 상태는 front==rear+1이다. X

(7) 원형 큐에서의 공백 상태는 front==0 && rear==0 이다. X

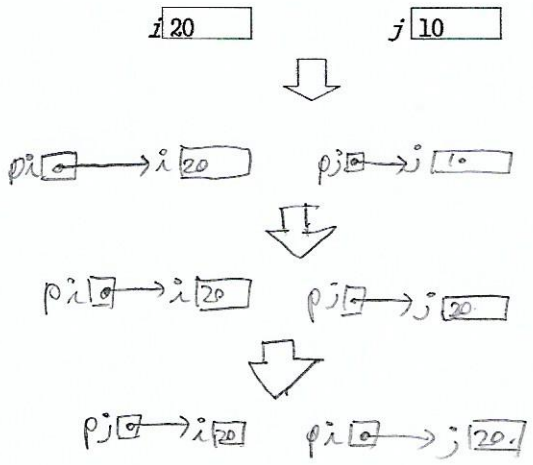
(8) 큐에 항목을 삽입하는 연산은 O(n) 이다. X

(9) 큐에 항목을 삭제하는 연산은 O(1) 이다. O

(10) 다항식 연산의 자료구조로 큐를 사용한다. X

2. 다음의 각각의 문장들을 수행한 다음에 변수들과 포인터들이 서로 연결된 모습을 그림으로 그려라.(10점)

```
void main() {
        int i=20, j=10;
        int *pi=&i, *pj=&j;
        *pj = *pi;
        pj = pi;
}
```

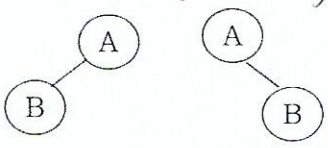


3. 다음 문장들 중 맞는 번호를 모두 고르시오.(10점) 1, 2, 5, 9 ☆

(1) 공집합도 이진트리에 속한다. O

(2) 링크 표현법에서는 부모 노드를 쉽게 알 수 있다. O

(3) 다음의 이진트리는 동일하다. X



(4) 배열 표현법은 완전 이진트리의 경우, 공간의 낭비가 심하다. X

(5) 포화 이진트리의 경우, 링크 표현법이 배열 표현법보다 메모리 공간을 더 많이 필요로 한다. O

(6) 노드의 개수가 n인 이진트리의 높이는 다를 수 있다. X

(7) 이진 탐색 트리에서는 오른쪽 자식 노드의 키값이 부모 노드의 키값보다 항상 크거나 같다. X
같은값은 왼쪽으로 보낸다

(8) 트리는 선형 자료 구조의 일종이다. X
↳선형X, 비선형 자료구조다!

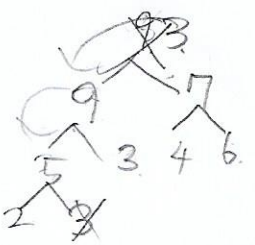
(9) 모든 포화 이진트리는 완전 이진트리이다. O

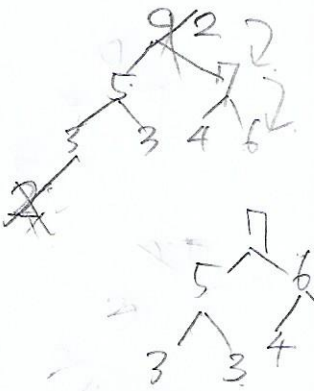
(10) 높이가 3인 이진트리에 존재할 수 있는 최대 노드의 개수는 8개이다. X

4. 다음 순서로 자료가 입력되었다고 가정하여 최대 힙을 구성한 후 2번의 삭제연산후의 배열 구조를 그리시오. 힙의 루트는 배열 인덱스 1번에 저장한다. 힙 구조는 그릴 필요 없고 배열만 완성하시오.(10점)

5, 3, 7, 2, 3, 4, 6, 9, 9 → 9개

0	1	2	3	4	5	6	7
	7	5	6	3	3	4	2





5. 다음 힙에 대한 물음에 답하시오.(5점)

(1) 히프 트리에서 노드가 삭제되는 위치는 어디인가?

루트 노드

(2) 히프가 배열로 표현될 수 있는 이유는 무엇인가?

완전 이진 트리이므로 각 노드에 번호를 붙일 수 있고 이 번호를 배열의 인덱스라고 생각하면 되기 때문이다.

(3) 히프 연산중에서 하나의 노드가 삽입되거나 삭제되는 시간은 무엇에 비례하는가?

히프 트리의 높이

(4) 노드가 18개인 히프의 높이는? 5

$\lceil \log(n+1) \rceil \Rightarrow \lceil \log 19 \rceil \Rightarrow$

or $\lceil \log(n) \rceil + 1 \Rightarrow \lceil \log 18 \rceil = 5$

(5) 다음 중 히프 정렬이 특히 유용하게 사용될 수 있는 경우는?

(1) 데이터 100개중에서 오름차순으로 20개만 뽑고자 할 때

(2) 비교적 데이터의 개수가 적을 때

(3) 정렬의 대상이 되는 레코드의 크기가 클 때

(4) 데이터가 역순으로 정렬되어 있을 때

6. 이진트리에서 높이를 구하는 함수를 완성하시오.(5점)

```
#define MAXIMUM(a, b) (a>b ? a : b)
int height(TreeNode *node)
{
  int height=0;
  if( node != NULL )
    //아래를 완성하시오.
    height = 1 + MAXIMUM(height(node->left), height(node->right));

  return height;
}
```

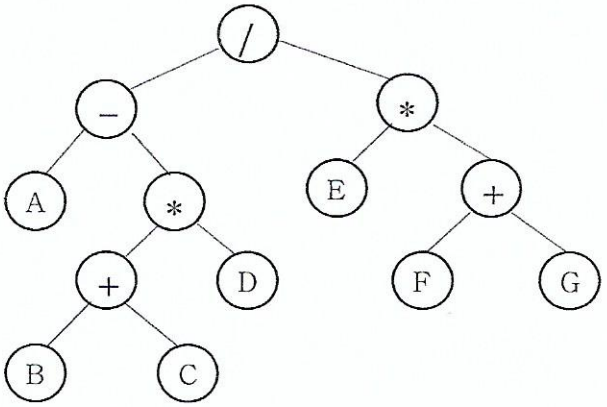
7. 단순 연결 리스트의 노드들을 노드 포인터 p로 탐색하고자 한다. p가 현재 가리키는 노드에서 다음 노드로 가려면 어떻게 하여야 하는가?(5점) 3

(1) p++; (2) p--; (3) p=p->link; (4) p=p->data;

8. 배열과 연결 리스트의 장단점을 간단하게 비교하시오.(5점)

- 연결 리스트(linked list) ↔ 배열(array)
- 구현은 복잡 ↔ 구현은 간단
- insert, delete 등의 매우 효율적 ↔ insert, delete 등의 비효율적임 (항목들의 이동이 많음).
- 최대 요소의 개수에 제한 없음 ↔ 최대 요소의 개수 제한 있음. (배열의 크기 때문)

9. 다음 수식 트리에 대하여 각각의 표기식으로 표현하시오.(10점)



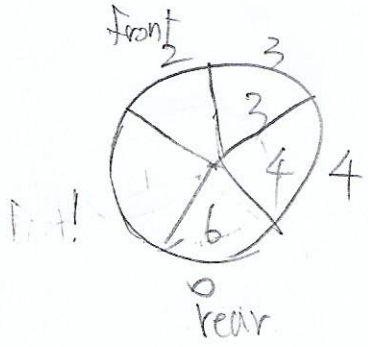
1) 전위 표기식: / - A * + B C D * E + F G

2) 중위 표기식 (A - ((B+C)*D) / E * (F+G)

3) 후위 표기식: A B C + D * E F G + * /

10. front와 rear의 초기값은 0이며 크기가 5인 원형 큐 A에 다음과 같이 삽입과 삭제가 되풀이 되었을 경우에 최종의 front와 rear의 값과 원형 큐의 내용을 나타내어라. 단 요소가 없을 경우는 ?를 쓰시오.(10점)

dequeue(A); →error
enqueue(A, 1);
enqueue(A, 2);
enqueue(A, 3);
enqueue(A, 4);
enqueue(A, 5); →error
dequeue(A);
enqueue(A, 6);
enqueue(A, 7); →error
dequeue(A);



front= rear=

0	1	2	3	4
6	?	?	3	4

11. 서브 트리에서의 가장 큰 키값을 가진 노드를 구하는 함수를 완성하시오. 또한 이진 탐색트리에서의 삭제시 3가지 경우를 설명하시오. (10점)

```
treePointer FindLarge(treePointer tree) {
    if (!tree) return NULL;
    //아래를 완성하시오.
    while (tree -> right = NULL) {
        if (tree -> right = NULL) break;
        else tree = tree -> right;
    }
    return tree;
}
```

- 삭제시의 3가지 경우

Case 1: 삭제할 노드가 단말 노드인 경우
Case 2: 삭제할 노드가 하나의 서브트리를 가지고 있는 경우
Case 3: 삭제할 노드가 두개의 서브트리를 가지고 있는 경우

12. 허프만 알고리즘을 사용하여 다음 표에 있는 글자들에 대한 허프만 코드를 구축하시오.(10점)

글자 :	A	B	C	D	E	F
빈도수:	2	3	7	5	2	3

Letter	Code
A	111
B	011
C	00
D	10
E	110
F	010