

과 목 명	자료구조와 알고리즘	2016년 1학기 중간고사	페이지	1
학 번		성 명		

1. 알고리즘에 소요되는 시간 복잡도를 나타내는 Big O 표현의 정의는 다음과 같다.

$f(n) = O(g(n))$ iff there exist positive constants c and n_0 such that $f(n) \leq c \cdot g(n)$ for all $n, n \geq n_0$

- 1) $O(1)$ 의 의미는 무엇인가?(2점)

상속시간으로 입력크기에 관계없이
항상 동일한 시간이 걸림

- 2) $O(n)$ 의 의미는 무엇인가?(3점)

선형시간을 의미하며
입력크기에 비례하며
수행시간이 소요된다.

2. 아래의 C 프로그램의 중간에 data가 5 개 정수 타입 요소로 구성된 배열로 사용될 수 있도록 동적 메모리를 할당받는 프로그램을 작성하시오.(5점)

```
int * data;
```

```
data = (int*) malloc(sizeof(int));
```

```
for(int i=0; i< 5; i++)
```

```
data[i] = i;
```

```
...
```

3. C 언어에서 다차원 배열(Array)은 열 우선 순서(Column Major Order)를 사용한다. char sales[1051][1200];로 정의된 배열에서 sales[0][0]의 주소가 320번지 일 경우 sales[12][127]의 주소는 무엇인가?(5점)

$320 + (12 \times 1200 + 127)$

4. factorial($n! = n \times (n-1)!, n \geq 1$) 값을 구하는 재귀호출 함수를 C 프로그램으로 작성하시오.(5점)

```
int factorial(int n){
```

```
if (n <= 1) return 1;
```

```
else
```


```
return (n * factorial(n-1));
```

```
}
```

5. 순서리스트(Ordered List)는 리스트들에 존재하는 원소들 사이에 순서를 가지는 리스트이다. 순서리스트를 단순 연결 리스트(Single Linked List)로 구현할 경우 특정 원소의 삽입과 제거에 소요되는 시간 복잡도를 Big O 표기법으로 나타내고, 그 이유를 기술하시오. (10점)

$O(n)$

삽입과 삭제 위치를 찾아야
하므로 선형 탐색.

과 목 명	자료구조와 알고리즘	2016년 1학기 중간고사	페이지	2
<p>6. <u>헤드노드가</u> 존재하는 <u>이중 연결 리스트</u>에서 삽입(insert())과 삭제(delete()) 함수를 C 프로그램으로 작성하시오.</p> <p>이중 연결 리스트에서는 empty list의 경우 헤드노드의 링크들이 헤드노드를 가르키며, empty list가 아닐 경우 헤드노드의 rlink는 첫 번째 요소 노드를, llink는 마지막 요소 노드를 가르킨다. 마찬가지로 첫 번째 요소 노드의 llink는 헤드 노드를, 마지막 요소 노드의 rlink는 헤드 노드를 가르킨다.(20점)</p> <pre>typedef struct DNode{ int data; struct DNode * llink; struct DNode * rlink; } DNode;</pre>  <pre>void insert(DNode * before, DNode * new_node){ // new_node를 before 노드 오른쪽에 삽입</pre> <div data-bbox="79 940 774 1937" style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <pre> new_node → rlink = before → rlink; new_node → llink = before; before → rlink → llink = new_node; before → rlink = new_node; </pre> </div> <pre> } </pre>		<pre>void delete(Dnode * h_node, DNode * r_node){ // r_node는 삭제되는 노드를 포인팅 rem // h_node는 리스트의 헤드노드를 포인팅 if (r_node == h_node) return; r_node → llink → rlink = r_node → rlink; r_node → rlink → llink = r_node → llink; free(r_node); }</pre>		

과 목 명	자료구조와 알고리즘	2016년 1학기 중간고사	페이지	3
<p>7. 원형 연결 리스트(Circular Linked List)가 단순 연결 리스트(Single Linked List)와 다른 점이 무엇이며, 원형 연결 리스트의 끝을 나타내는 방법을 기술하시오.(10점)</p> <p>원형연결리스트는 마지막노드가 헤드노드를가리킴</p> <p>헤드노드 사용</p>		<pre> int dequeue(QueueType * q){ if(!q->empty(q)) printf("큐 비어") exit(1); q->front=(q->front+1)%MAX return q->que[q->front]; } </pre>		
<p>8. 배열을 이용하여 원형 큐를 구현할 때, 원형 큐에서 삽입(enqueue())과 삭제(dequeue()) 함수를 C 언어로 완성하시오. 큐는 초기화되었다고 가정한다.(15점)</p> <pre> typedef struct { int que[100]; int front, rear; } QueueType; void enqueue(QueueType * q, int e){ </pre>		<pre> if(!q->full(q)) printf("큐 꽂음") exit(1); q->rear=(q->rear+1)%100; return q->que[q->rear]=element; } </pre>		
<p>9. 다음 프로그램 수행 결과를 정확하게 나타내시오 (5점)</p> <pre> int main(int ac, char * av[]){ int data[] = {0,1,2,3,4,5,6,7,8,9,10,11,12}; int *ptr = &data[3]; printf("The value is %d\n", ptr[5]); } </pre>		<p>ptr+5</p>		

과 목 명	자료구조와 알고리즘	2016년 1학기 중간고사	페이지	4
<p>10. 스택을 이용하여 후위표기식의 값을 구하는 C 함수 <code>int eval(char exp[])</code>를 작성하시오. 매개변수 <code>exp</code>에는 후위표기식이 스트링으로 주어진다. 입력 문자는 숫자(0~9)와 스페이스('Wb') 그리고 사칙연산 기호만 존재하며, 숫자는 다단위 숫자가 가능하며 0으로 시작하지 않는다. 스택은 초기화되어 있으며, 함수 <code>pop()</code>과 <code>push()</code>는 제공된다고 가정한다(20점)</p> <p>입력 예: 125 25 + 30 * , 리턴 값: 4500</p> <pre> typedef struct { int stack[1000]; int top; } StackType; extern int pop(StackType * s); extern void push(StackType * s, int v); int eval(char expr){ </pre>				
<pre> StackType s; s.top = -1; int num = 0; int i = 0; while (expr[i] != '\0') { if (expr[i] == ' ') i++; continue; if (isdigit(expr[i])) num = 0; while (isdigit(expr[i])) { num = num * 10 + pop(&s, num); } ege int b = pop(&s); int a = pop(&s); a+b a-b a*b a/b i++; return pop(&s); </pre>				

}