

연습문제

EXERCISE

- 01 2개의 정수를 서로 교환하는 알고리즘을 의사 코드로 작성해보자.
- 02 사용자로부터 받은 2개의 정수 중에서 더 큰 수를 찾는 알고리즘을 의사코드로 작성해보자.
- 03 1부터 n 까지의 합을 계산하는 알고리즘을 의사 코드로 작성해보자.
- 04 Set(집합) 추상 자료형을 정의하라. 다음과 같은 연산자들을 포함시켜라.
Create, Insert, Remove, Is_In, Union, Intersection, Difference.
- 05 Boolean 추상 자료형을 정의하고 다음과 같은 연산자들을 포함시켜라.
And, Or, Not, Xor
- 06 다음과 같은 코드의 시간 복잡도는? 여기서 n 이 프로그램의 입력이라고 가정하자.

```
for(i = 1; i < n; i *= 2)
    printf("Hello");
```
- 07 다음과 같은 코드의 시간 복잡도는? 여기서 n 이 프로그램의 입력이라고 가정하자.

```
for(i = 0; i < n; i++)
    for(j = 1; j < n; j *= 2)
        printf("Hello");
```
- 08 시간 복잡도 함수 $n^2 + 10n + 8$ 를 빅오 표기법으로 나타내면?
 (1) $O(n)$ (2) $O(n \log_2 n)$
 (3) $O(n^2)$ (4) $O(n^2 \log_2 n)$
- 09 시간 복잡도 함수가 $7n + 10$ 이라면 이것이 나타내는 것은 무엇인가?
 (1) 연산의 횟수 (2) 프로그램의 수행시간
 (3) 프로그램이 차지하는 메모리의 양 (4) 입력 데이터의 총개수
- 10 $O(n^2)$ 의 시간복잡도를 가지는 알고리즘에서 입력의 개수가 2배로 되었다면 실행시간은 어떤 추세로 증가하는가?
 (1) 변함없다. (2) 2배
 (3) 4배 (4) 8배

11 $f(n)$ 에 대하여 엄격한 상한을 제공하는 표기법은 무엇인가?

- (1) 빅오메가 (2) 빅오
(3) 빅세타 (4) 존재하지 않는다.

12 다음의 빅오표기법들을 수행시간이 적게 걸리는 것부터 나열하라.

$O(1)$ $O(n)$ $O(\log n)$ $O(n^2)$ $O(n \log n)$ $O(n!)$ $O(2^n)$

13 두 함수 $30n+4$ 와 n^2 를 여러 가지 n 값으로 비교하라. 언제 $30n+4$ 가 n^2 보다 작은 값을 갖는지를 구하라. 그래프를 그려보라.

14 다음은 실제로 프로그램의 수행시간을 측정하여 도표로 나타낸 것이다. 도표로부터 이 프로그램의 시간 복잡도를 예측하여 빅오 표기법으로 나타내라.

입력의 개수 n	수행시간 (초)
2	2
4	8
8	25
16	63
32	162

15 빅오표기법의 정의를 사용하여 다음을 증명하라.

$$5n^2 + 3 = O(n^2)$$

16 빅오 표기법의 정의를 이용하여 $6n^2 + 3n$ 이 $O(n)$ 이 될 수 없음을 보여라.

17 배열에 정수가 들어 있다고 가정하고 다음의 작업의 최악, 최선의 시간복잡도를 빅오 표기법으로 말하라.

- (1) 배열의 n 번째 숫자를 화면에 출력한다.
(2) 배열안의 숫자 중에서 최소값을 찾는다.
(3) 배열의 모든 숫자를 더한다.

연습문제

EXERCISE

01 팩토리얼을 계산하는 순환호출 함수 factorial에서 매개 변수로 5를 주었다면 최대 몇 개의 factorial 함수의 활성 레코드가 동시에 존재할 수 있는가?

02 순환 호출을 하였을 경우에 활성 레코드들이 저장되는 위치는 어디인가?

- | | |
|---------------|--------|
| (1) 순환호출 함수내부 | (2) 변수 |
| (3) 배열 | (4) 스택 |

03 다음 중 활성 레코드에 저장되지 않는 것은 무엇인가?

- | | |
|-------------|----------------------|
| (1) 매개변수의 값 | (2) 함수호출이 끝나고 복귀할 주소 |
| (3) 지역변수 | (4) 순환호출의 순차번호 |

04 하나의 함수가 호출할 수 있는 순환호출의 개수는?

- | | |
|-----------------|---------|
| (1) 1번 | (2) 2번 |
| (3) 스택이 허용하는 한도 | (4) 무제한 |

05 다음의 순환호출 함수에서 잘못된 점은 무엇인가?

```
int recursive(int n)
{
    if( n==1 ) return 0;
    return n*recursive(n);
}
```

06 다음의 순환호출 함수에서 잘못된 점은 무엇인가?

```
int recursive(int n)
{
    printf("recursive(%d)\n", n);
    return n*recursive(n-1);
}
```

- 07 다음 함수를 sum(5)로 호출하였을 때, 화면에 출력되는 내용과 함수의 반환값을 구하라.

```
int sum(int n)
{
    printf("%d\n", n);
    if( n<1) return 1;
    else return( n*sum(n-1) );
}
```

- 08 다음 함수를 recursive(5)로 호출하였을 때, 화면에 출력되는 내용과 함수의 반환값을 구하라.

```
int recursive(int n)
{
    printf("%d\n", n);
    if( n<1) return 2;
    else return( 2*recursive(n-1)+1 );
}
```

- 09 다음 함수를 recursive(10)로 호출하였을 때, 화면에 출력되는 내용과 함수의 반환값을 구하라.

```
int recursive(int n)
{
    printf("%d\n", n);
    if( n<1) return -1;
    else return( recursive(n-3)+1 );
}
```

- 10 다음 함수를 recursive(5)로 호출하였을 때, 화면에 출력되는 내용을 쓰시오.

```
int recursive(int n)
{
    if(n != 1) recursive(n-1);
    printf("%d\n", n);
}
```

- 11 다음 함수에서 asterisk(5)를 호출할 때 출력되는 *의 갯수는?

```
void asterisk(int i)
{
    if( i > 1 ){
        asterisk(i/2);
        asterisk(i/2);
    }
    printf("*");
}
```

<CH2>

- 12 다음과 같은 함수를 호출하고 "recursive" 문자열을 입력한 다음, 엔터키를 눌렀다면 화면에 출력되는 것은?

```
unknown()
{
    int ch;
    if( (ch=getchar()) != '\n')
        unknown();
    putchar();
}
```

- 13 다음을 계산하는 순환적인 프로그램을 작성하시오.

$$1 + 2 + 3 + \dots + n$$

- 14 다음을 계산하는 순환적인 프로그램을 작성하시오.

$$1 + 1/2 + 1/3 + \dots + 1/n$$

- 15 순환 호출되는 것을 이해하기 위하여 fib 함수를 다음과 같이 바꾸어서 실행하여 보라. fib(6)을 호출할 때 화면에 출력되는 내용을 쓰시오.

```
int fib(int n)
{
    printf("fib(%d) is called\n", n);
    if( n==0 ) return 0;
    if( n==1 ) return 1;
    return (fib(n-1) + fib(n-2));
}
```

- 16 다음의 순환적인 프로그램을 반복 구조를 사용한 비순환적 프로그램으로 바꾸시오.

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

- 17 이항계수(binomial coefficient)를 계산하는 순환함수를 작성하라. 이항계수는 다음과 같이 순환적으로 정의된다. 반복함수로도 구현해보라.

$${}_nC_k = \begin{cases} {}_{n-1}C_{k-1} + {}_{n-1}C_k & \text{if } 0 < k < n \\ 1 & \text{if } k=0 \text{ or } k=n \end{cases}$$

- 18 Ackermann 함수는 다음과 같이 순환적으로 정의된다.

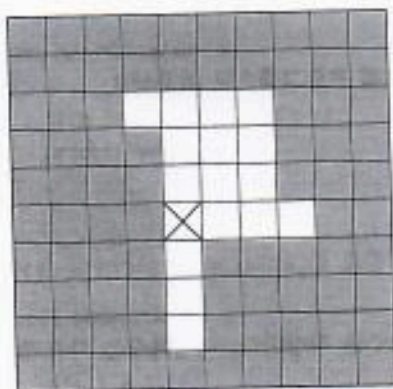
$$\begin{aligned} A(0, n) &= n + 1; \\ A(m, 0) &= A(m-1, 1) \\ A(m, n) &= A(m-2, A(m, n-1)) \quad m, n \geq 1 \end{aligned}$$

- (a) $A(3, 2)$ 와 $A(2, 3)$ 의 값을 구하시요.
 (b) Ackermann 함수를 구하는 순환적인 프로그램을 작성하시요.
 (c) 위의 순환적인 프로그램을 for, while, do와 같은 반복구조를 사용한 비순환적 프로그램으로 바꾸시오.
- 19 본문의 순환적인 피보나치 수열 프로그램과 반복적인 피보나치 수열 프로그램의 수행 시간을 측정하여 비교하라. 어떤 결론을 내릴 수 있는가?

20 순환호출에서는 순환호출을 할때마다 문제의 크기가 작아져야 한다.

- (1) 팩토리얼 계산 문제에서 순환호출이 일어날 때마다 문제가 어떻게 작아지는가?
 (2) 하노이의 탑에서 순환호출이 일어날 때마다 문제의 어떻게 작아지는가?

- 21 컴퓨터 그래픽에서의 영역 채우기 알고리즘은 순환 기법을 사용한다. 영역 채우기란 다음과 같은 흰색 영역이 있을 때 이 영역을 특정한 색으로 채우는 것이다. 여기서는 이 영역 안쪽을 검정색으로 채운다고 가정해보라. 이런 경우에는 순환 호출을 어떻게 사용할 수 있을까? 2차원 배열이 다음과 같이 되어 있다고 가정하고 영역안의 한 점의 좌표가 주어졌을 경우에 안쪽을 채우는 순환 호출 함수를 작성하여 보라. [그림 2-16]의 ×로 표시된 픽셀이 시작 픽셀이다.



(a) 영역 채우기 대상 도형

2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2
2	2	2	0	0	0	0	2	2	2
2	2	2	2	0	0	0	2	2	2
2	2	2	2	0	0	0	2	2	2
2	2	2	2	×	0	0	0	2	2
2	2	2	2	0	2	2	2	2	2
2	2	2	2	0	2	2	2	2	2
2	2	2	2	0	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2

(b) 2차원 배열을 이용하여 (a)를 표현

[그림 2-16] 영역 채우기

Hint 위의 문제를 해결하려면 먼저 위의 [그림 2-16] (a)를 2차원 배열을 이용하여 (b)와 같이 나타낸다. 노란색은 2로, 흰색은 0으로 영역을 표시한 다음, 0의 값을 갖는 픽셀을 전부 1로 바꾸면 되는 것이다. 다음 프로그램의 빈칸을 채운 다음, 수행시켜서 어떤 순서대로 채워지는지를 살펴본다.

```
#define WHITE 0
#define BLACK 1
```

01 int a[10][20]에서 배열이 차지하는 메모리 공간의 크기는 얼마인가? int형은 4바이트라고 하자.

- (1) 200바이트 (2) 400바이트
(3) 600바이트 (4) 800바이트

02 float a[100]으로 선언된 배열의 시작 주소를 1000번지라고 할 때, 배열의 10번째 요소의 주소는 몇 번지인가?

- (1) 1100번지 (2) 1200번지
(3) 1300번지 (4) 1400번지

03 다음 배열 중에서 크기가 가장 큰 배열은?

- (1) int array1[10]; (2) double array2[10];
(3) char array3[40]; (4) float array4[10];

04 크기가 10인 배열 two[]를 선언하고 여기에 2의 제곱 값들을 저장해보자. 즉 배열의 첫 번째 요소에는 2를 저장하고 두 번째 요소에는 4를 저장한다. 마지막 요소에는 512를 저장한다. for 루프를 이용하여 two[] 배열의 전체 요소의 값을 출력하는 프로그램을 작성하라.

05 person이라는 구조체를 만들어보자. 이 구조체에는 문자 배열로 된 이름, 사람의 나이를 나타내는 정수 값, 각 개인의 월급을 나타내는 float 값 등이 변수로 들어가야 한다.

06 C언어의 typedef을 이용하여 complex라고 하는 새로운 자료형을 정의하라. complex 자료형은 구조체로서 float형인 real 변수와 역시 float형인 imaginary 변수를 갖는다. complex 자료형으로 변수 c1과 c2를 선언하여보라.

07 복소수를 구조체로 표현해보자. 복소수 a와 복소수 b를 받아서 a+b를 계산하는 함수를 작성해보자. 함수는 구조체를 반환할 수 있다. 알다시피 복소수는 $real + imag \cdot i$ 와 같은 형태를 갖는다.

```
Complex complex_add(Complex a, Complex b) {
    ...
}
```

08 크기가 n인 배열 array에서 임의의 위치 loc에 정수 value를 삽입하는 함수 insert()를 작성하라. 정수가 삽입되면 그 뒤에 있는 정수들은 한 칸씩 뒤로 밀려야 한다. 현재 배열에 들어있는 원소의 개수는 items개라고 하자. (여기서 $items < n$ 라고 가정)

<CH3>

```
void insert(int array[], int loc, int value){  
    ....  
}
```

09 앞의 문제에서 구현한 insert() 함수의 시간 복잡도는?

10 크기가 n인 배열 array에서 임의의 위치 loc에 있는 정수를 삭제하는 함수 delete()를 작성하라. 정수가 삭제되면 그 뒤에 있는 정수들은 한 칸씩 앞으로 이동하여야 한다. 현재 배열에 들어 있는 원소의 개수는 items개라고 하자. (여기서 items < n라고 가정)

```
int delete(int array[], int loc){  
    ....  
}
```

11 앞의 문제에서 구현한 delete() 함수의 시간 복잡도는?

12 1개의 정수와 최대 크기가 20인 문자열로 이루어진 구조체를 저장할 수 있도록 동적 메모리를 할당받고 여기에 정수 100과 문자열 "just testing"을 저장한 다음, 동적 메모리를 반환하는 프로그램을 작성하라.

연습문제

EXERCISE

01 이진 탐색 알고리즘의 특징이 아닌 것은?(기사 시험 기출 문제)

- ① 탐색 효율이 좋고 탐색 시간이 적게 소요된다.
- ② 검색할 데이터가 정렬되어 있어야 한다.
- ③ 피보나치 수열에 따라 다음에 비교할 대상을 선정하여 검색한다.
- ④ 비교를 거듭할 때마다 검색 대상이 되는 데이터의 수가 절반으로 줄어든다.

02 키 값 28을 가지고 아래의 리스트를 탐색할 때 다음의 탐색 방법에 따른 탐색 과정을 그리고 탐색 시에 필요한 비교 연산 횟수를 구하여라.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
8	11	12	15	16	19	20	23	25	28	29	31	33	35	38	40

- (1) 순차 탐색
- (2) 이진 탐색
- (3) 보간 탐색

03 AVL 트리에서 회전은 언제 이루어지는가?

- (1) 삽입 전
- (2) 삽입 후
- (3) 탐색 중
- (4) 탐색 후

04 정렬된 100,000,000개의 원소가 있다. 이진 탐색 알고리즘을 사용했을 때 최악의 경우에 대하여 비교 횟수를 구하라.

05 데이터 (60, 50, 20, 80, 90, 70, 55, 10, 40, 35)를 차례대로 삽입하면서 다음과 같은 균형트리를 구축하는 과정을 그림으로 설명하고 이들 3가지의 트리를 사용한 결과를 서로 비교하라.

- (a) 이진탐색트리
- (b) AVL 트리
- (c) 2-3 트리

06 데이터 (10, 20, 30, 40, 50, 60, 70, 80, 90, 100)를 차례대로 삽입했을 때의 결과 트리를 그려라. 어떤 트리가 탐색을 가장 효율적으로 수행하는가?

- (a) 이진탐색트리
- (b) AVL 트리
- (c) 2-3 트리