

Hanyang Univ.

자료구조론 2주차 실습

강진영

CONTENTS

1

포인터 실습

2

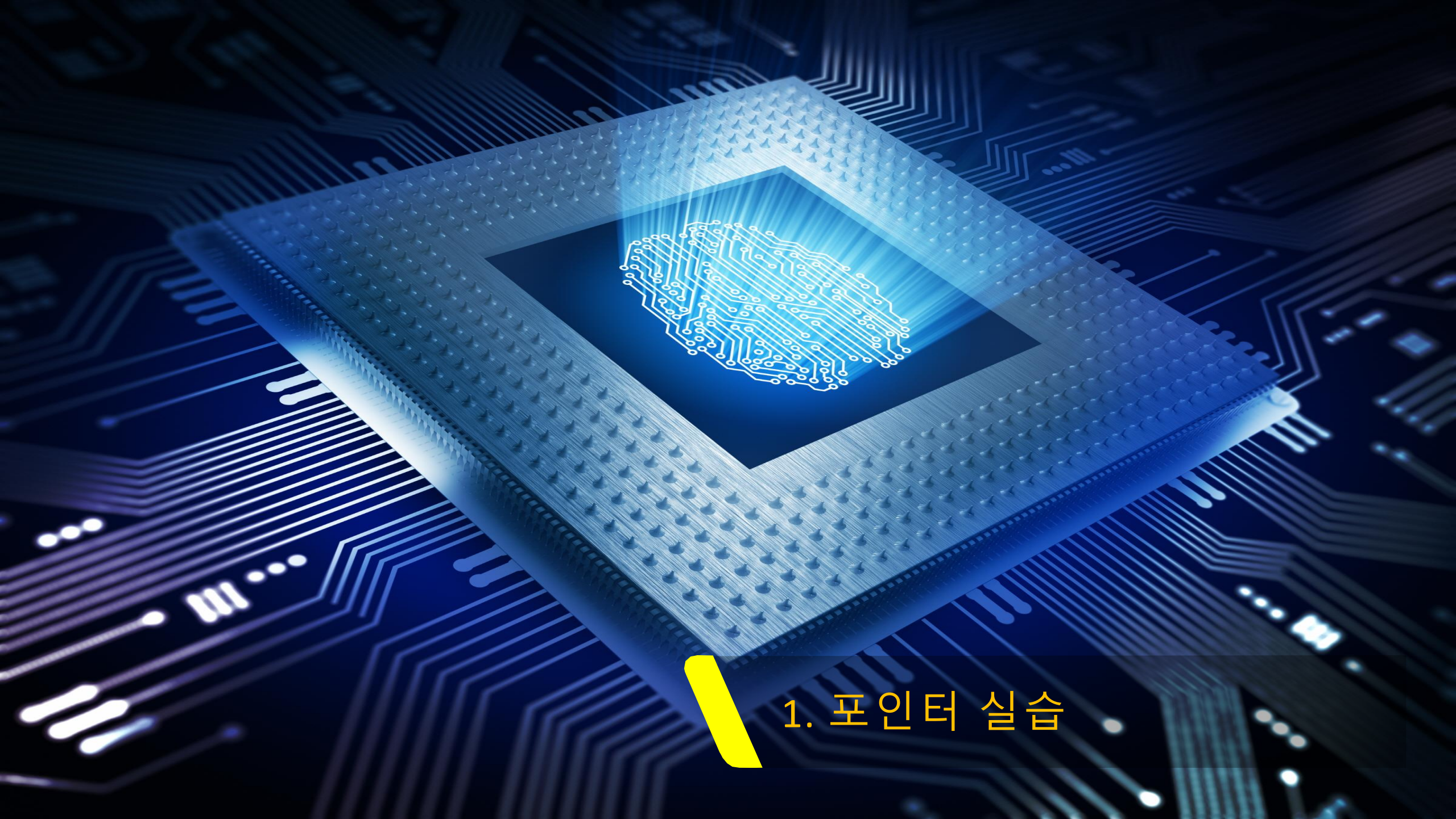
알고리즘 성능 측정 실습

3

팩토리얼 실습

4

배열 퀴즈



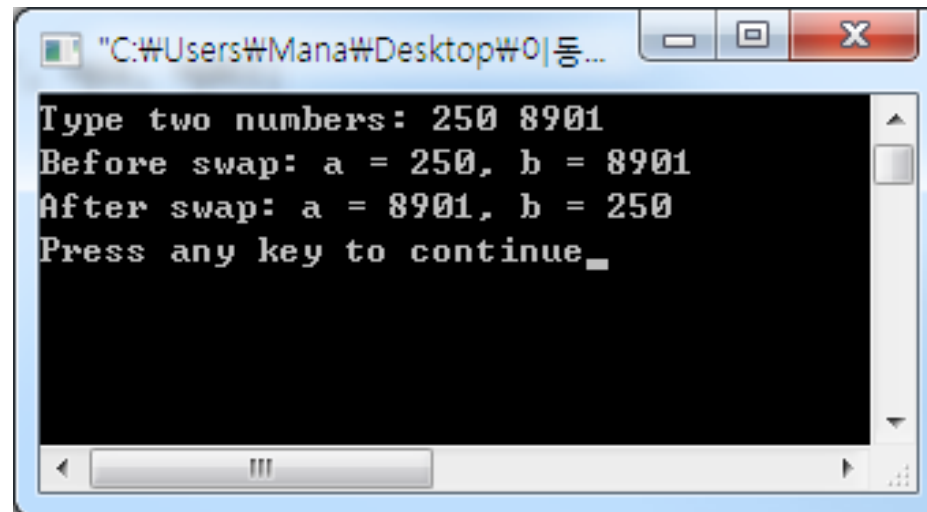
1. 포인터 실습

Contents

- 1. 포인터 실습
- 2. 성능 측정 실습
- 3. 팩토리얼 실습
- 4. 배열 실습

동적 메모리 할당 및 포인터

- 동적 메모리 할당 (malloc) 함수 이용 실습
 - 두 개의 포인터 변수를 선언 (예: *pa, *pb)
 - 각각의 포인터에 동적으로 메모리를 할당하고 숫자를 입력
 - Swap 함수를 구현하여 swap 전과 swap 후의 값을 비교



```
"C:\Users\Mana\Desktop\이동..."
Type two numbers: 250 8901
Before swap: a = 250, b = 8901
After swap: a = 8901, b = 250
Press any key to continue_
```


1. 포인터 실습

Contents

1. 포인터 실습

2. 성능 측정 실습

3. 팩토리얼 실습

4. 배열 실습

Swap 실습

```
#include <stdio.h>
#include <malloc.h>
```

```
void swap_own (int *a, int *b);
```

```
void main (void)
{
```

```
    int *pa, *pb;
```

```
    pa = (int *) malloc (sizeof(int));
    pb = (int *) malloc (sizeof(int));
```

```
    printf ("Type two numbers: ");
    scanf ("%d %d", pa, pb);
```

```
    printf ("Before swap: a = %d, b = %d\n", *pa,
    *pb);
```

```
    swap_own (pa, pb);
```

```
    printf ("After swap: a = %d, b = %d\n", *pa, *pb);
```

```
    free (pa);
    free (pb);
```

```
}
```

```
void swap_own (int *a, int *b)
```

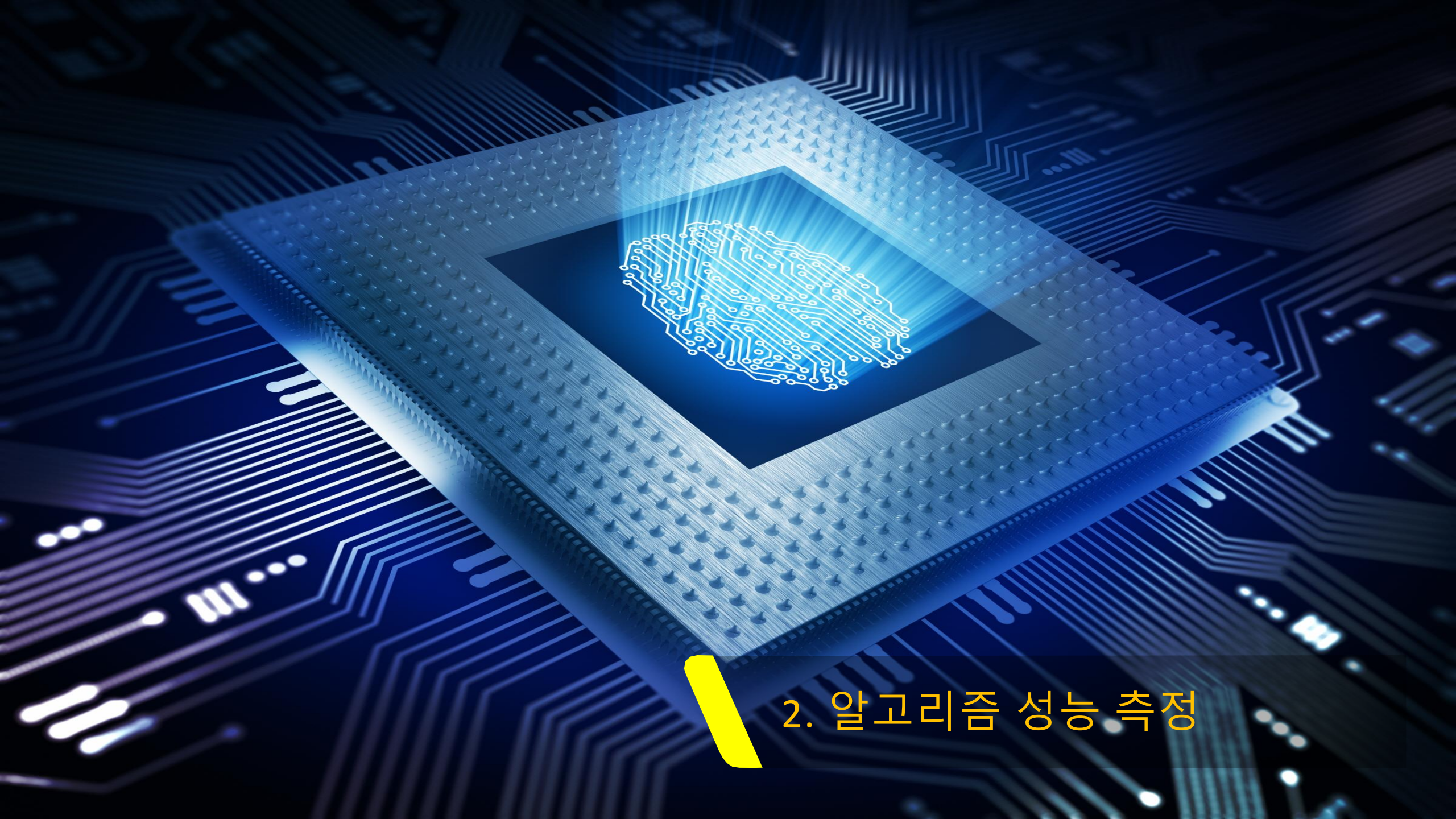
```
{
```

```
    *a = *a + *b;
```

```
    *b = *a - *b;
```

```
    *a = *a - *b;
```

```
}
```



2. 알고리즘 성능 측정

Contents

1. 포인터 실습

2. 성능 측정 실습

3. 팩토리얼 실습

4. 배열 실습

알고리즘 성능 분석

- 수행 시간 측정
 - 입력 받은 반복 횟수만큼 자기 자신을 곱하는 프로그램
 - `Result * = num;`
 - `int`형과 `float`형의 수행시간을 비교
- 수행시간 측정 함수
 - `clock_t clock(void);`
 - `clock` 함수는 호출되었을 때의 시스템 시각을 `CLOCKS_PER_SEC` 단위로 반환

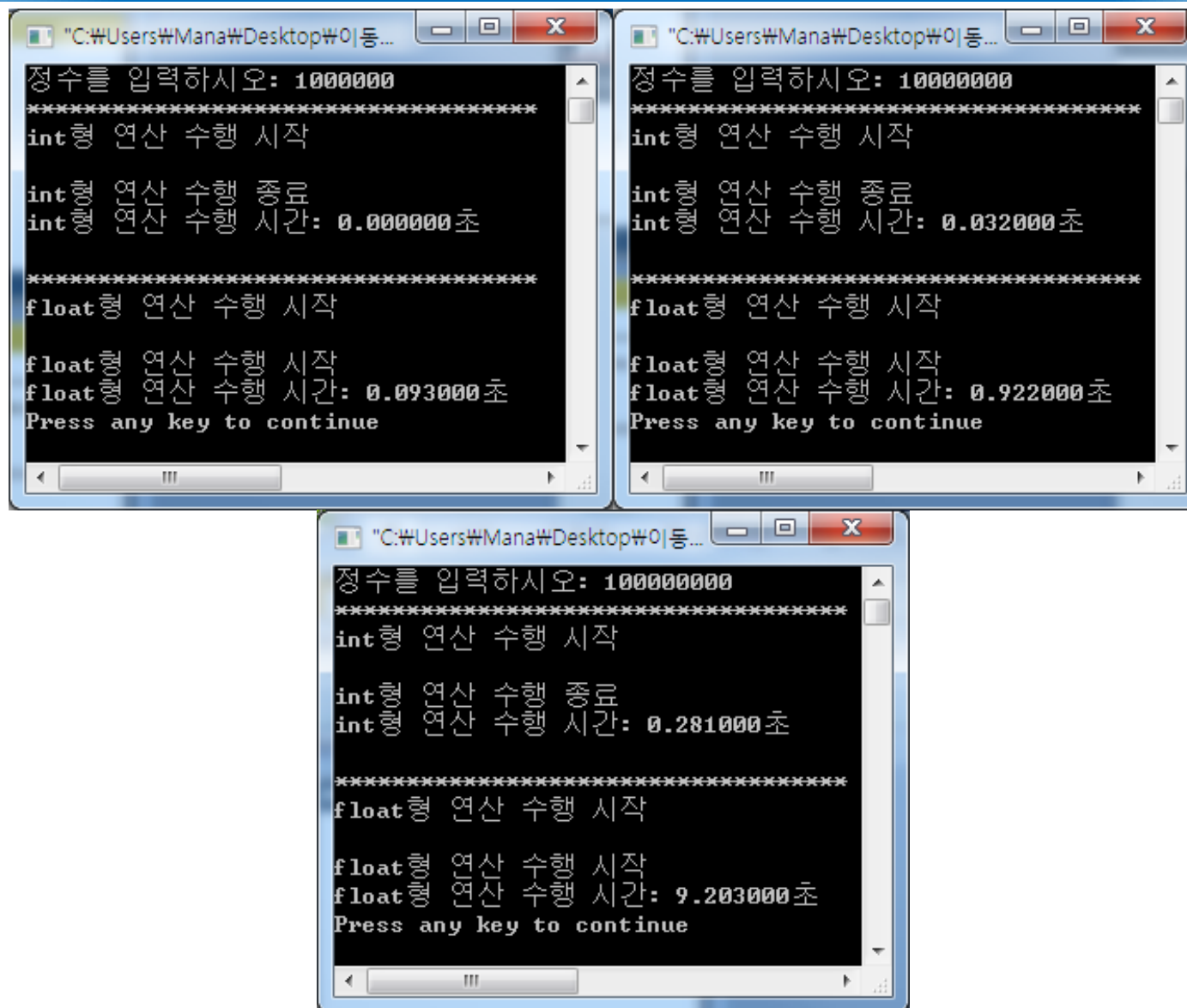
```
void main( void )
{
    clock_t start, finish;
    double duration;
    start = clock();
    // 수행시간을 측정하고자 하는 코드....
    // ....
    finish = clock();
    duration = (double)(finish - start) / CLOCKS_PER_SEC;
    printf("%f 초입니다.\n", duration);
}
```

2. 성능 측정 실습

Contents

1. 포인터 실습
2. 성능 측정 실습
3. 팩토리얼 실습
4. 배열 실습

알고리즘 성능 분석



```
"C:\Users\Mana\Desktop\이동..."
정수를 입력하시오: 10000000
*****
int형 연산 수행 시작

int형 연산 수행 종료
int형 연산 수행 시간: 0.000000초

*****
float형 연산 수행 시작

float형 연산 수행 시작
float형 연산 수행 시간: 0.093000초
Press any key to continue

"C:\Users\Mana\Desktop\이동..."
정수를 입력하시오: 100000000
*****
int형 연산 수행 시작

int형 연산 수행 종료
int형 연산 수행 시간: 0.032000초

*****
float형 연산 수행 시작

float형 연산 수행 시작
float형 연산 수행 시간: 0.922000초
Press any key to continue

"C:\Users\Mana\Desktop\이동..."
정수를 입력하시오: 100000000
*****
int형 연산 수행 시작

int형 연산 수행 종료
int형 연산 수행 시간: 0.281000초

*****
float형 연산 수행 시작

float형 연산 수행 시작
float형 연산 수행 시간: 9.203000초
Press any key to continue
```

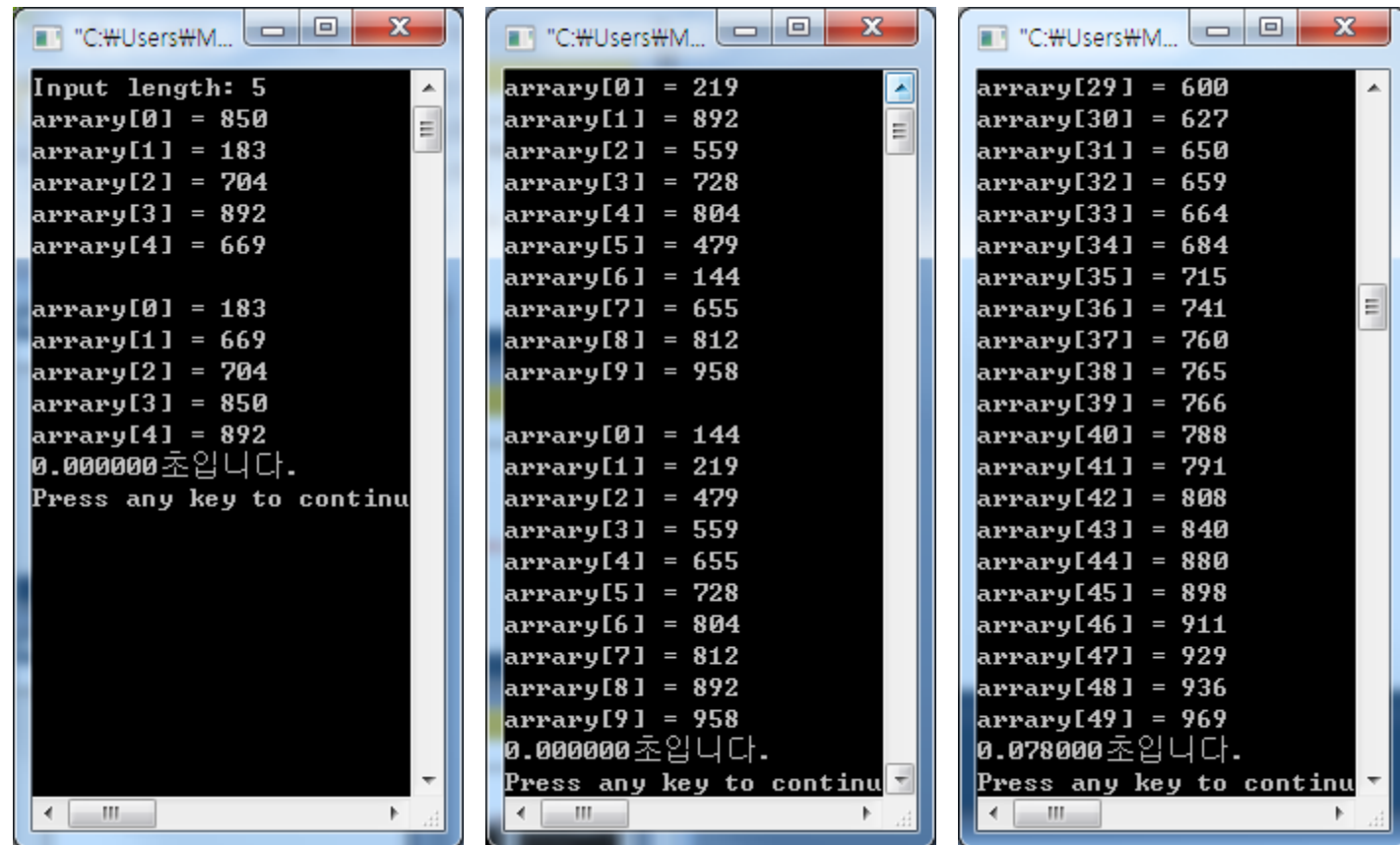

2. 성능 측정 실습

Contents

1. 포인터 실습
2. 성능 측정 실습
3. 팩토리얼 실습
4. 배열 실습

Insertion Sort 수행 시간 측정

- 배열에 임의의 random number를 생성/저장한 후 정렬
- Clock 함수를 이용하여 수행시간을 측정



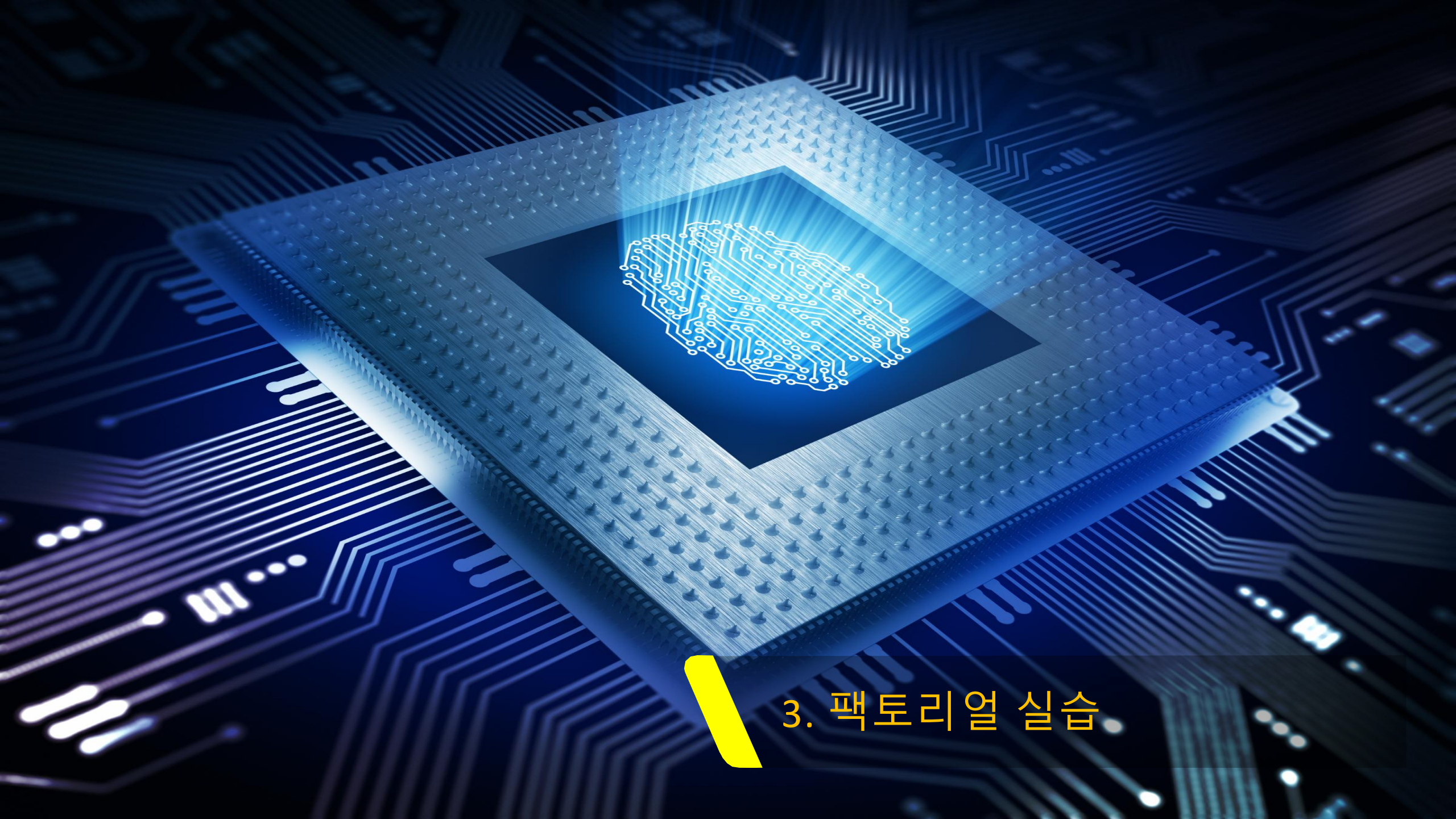
```
Input length: 5
array[0] = 850
array[1] = 183
array[2] = 704
array[3] = 892
array[4] = 669

array[0] = 183
array[1] = 669
array[2] = 704
array[3] = 850
array[4] = 892
0.000000초입니다.
Press any key to continu
```

```
array[0] = 219
array[1] = 892
array[2] = 559
array[3] = 728
array[4] = 804
array[5] = 479
array[6] = 144
array[7] = 655
array[8] = 812
array[9] = 958

array[0] = 144
array[1] = 219
array[2] = 479
array[3] = 559
array[4] = 655
array[5] = 728
array[6] = 804
array[7] = 812
array[8] = 892
array[9] = 958
0.000000초입니다.
Press any key to continu
```

```
array[29] = 600
array[30] = 627
array[31] = 650
array[32] = 659
array[33] = 664
array[34] = 684
array[35] = 715
array[36] = 741
array[37] = 760
array[38] = 765
array[39] = 766
array[40] = 788
array[41] = 791
array[42] = 808
array[43] = 840
array[44] = 880
array[45] = 898
array[46] = 911
array[47] = 929
array[48] = 936
array[49] = 969
0.078000초입니다.
Press any key to continu
```



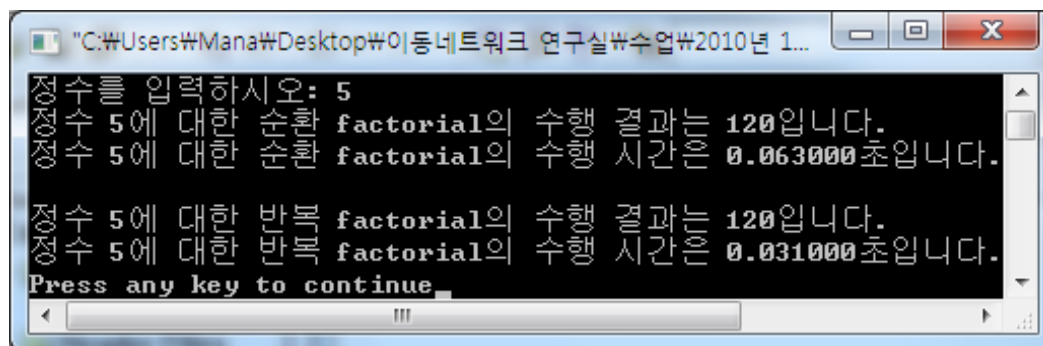
3. 팩토리얼 실습

Contents

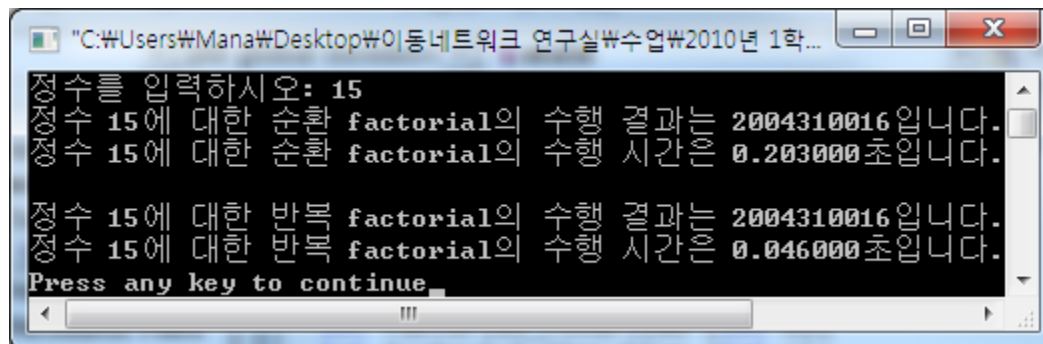
1. 포인터 실습
2. 성능 측정 실습
3. 팩토리얼 실습
4. 배열 실습

재귀함수 실습 (Feat. Factorial)

- Factorial 계산 프로그램
 - Factorial을 recursion과 iteration으로 구현
 - 수행시간을 비교 (수행 시간이 짧으므로 100만 번을 반복)



```
"C:\Users\Mana\Desktop\이동네트워크 연구실\수업\2010년 1학...  
정수를 입력하시오: 5  
정수 5에 대한 순환 factorial의 수행 결과는 120입니다.  
정수 5에 대한 순환 factorial의 수행 시간은 0.063000초입니다.  
  
정수 5에 대한 반복 factorial의 수행 결과는 120입니다.  
정수 5에 대한 반복 factorial의 수행 시간은 0.031000초입니다.  
Press any key to continue.
```



```
"C:\Users\Mana\Desktop\이동네트워크 연구실\수업\2010년 1학...  
정수를 입력하시오: 15  
정수 15에 대한 순환 factorial의 수행 결과는 2004310016입니다.  
정수 15에 대한 순환 factorial의 수행 시간은 0.203000초입니다.  
  
정수 15에 대한 반복 factorial의 수행 결과는 2004310016입니다.  
정수 15에 대한 반복 factorial의 수행 시간은 0.046000초입니다.  
Press any key to continue.
```

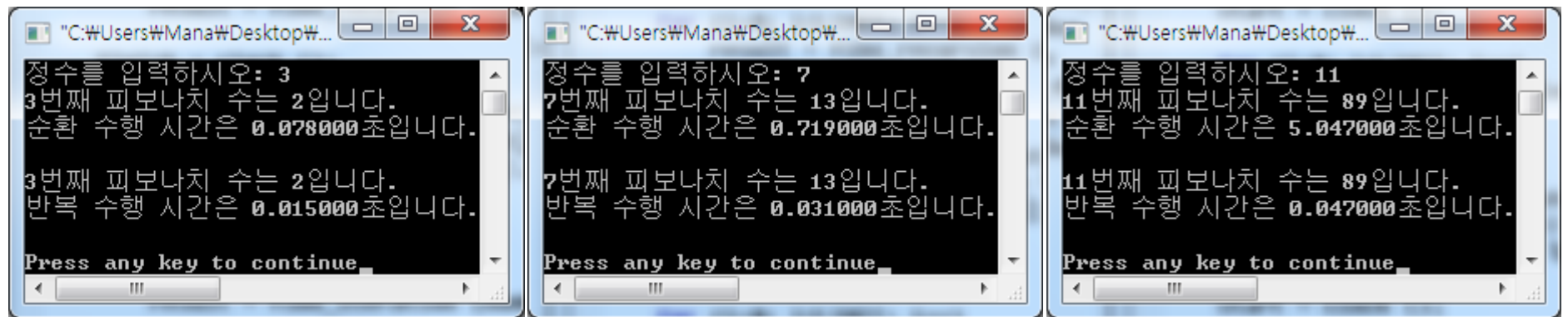

Contents

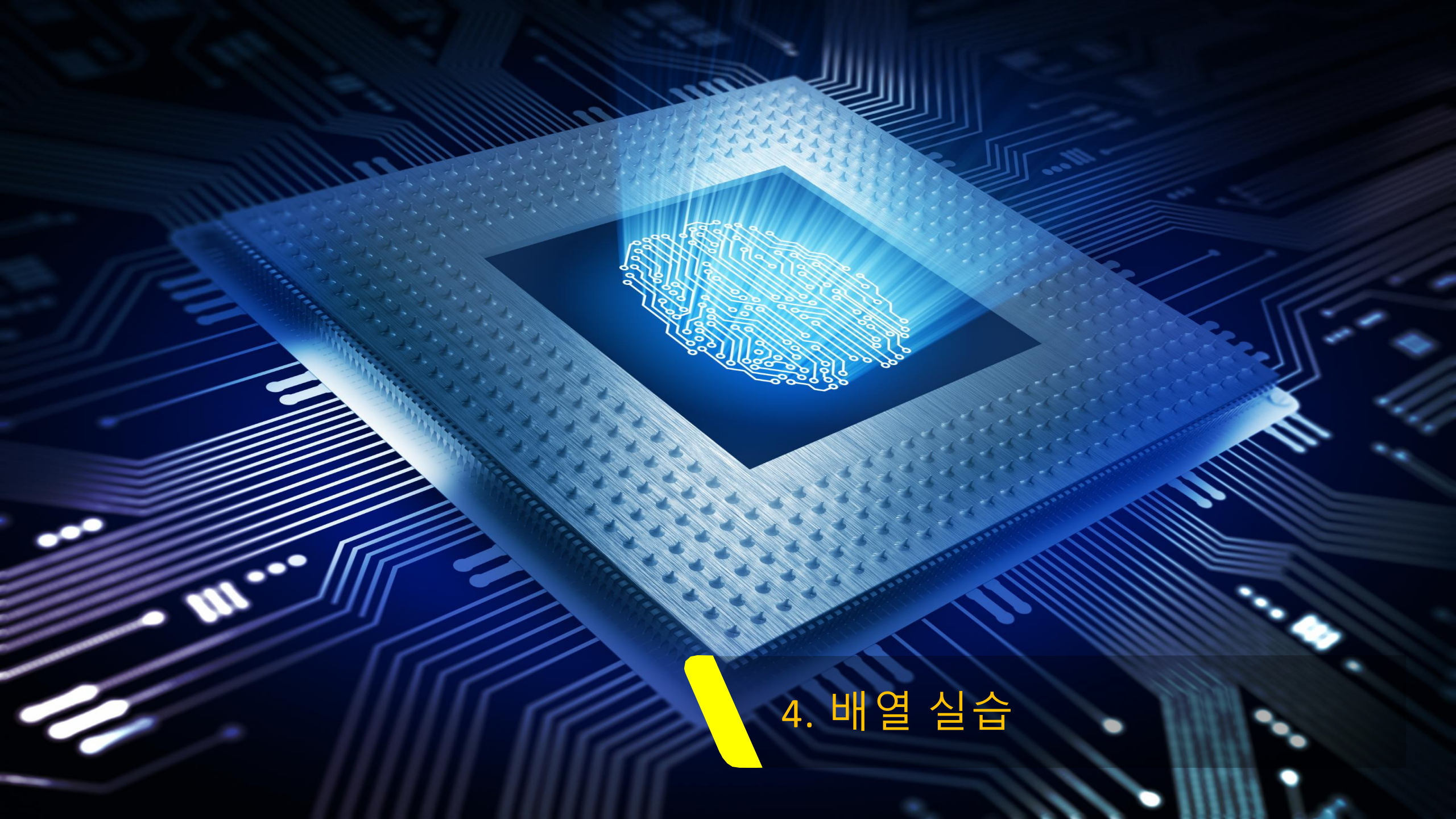
1. 포인터 실습
2. 성능 측정 실습
3. 팩토리얼 실습
4. 배열 실습

재귀함수 실습 (Feat. Factorial)

- 피보나치 수열
 - 입력 받은 정수(n) 번째 피보나치 수를 출력하는 프로그램
 - Recursion과 iteration의 수행 시간 비교 (100만 번 반복)

$$fib(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ fib(n-2) + fib(n-1) & \text{otherwise} \end{cases}$$





4. 배열 실습

4. 배열 실습

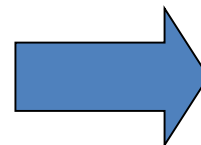
Contents

1. 포인터 실습
2. 성능 측정 실습
3. 팩토리얼 실습
4. 배열 실습

배열 실습하기 (1/2)

- 3x3크기의 2차원 배열의 행과 열을 바꾸는 프로그램
 - 배열의 값을 임의로 입력

(0,0)	(0,1)	(0,2)
10	12	14
(1,0)	(1,1)	(1,2)
16	18	20
(2,0)	(2,1)	(2,2)
22	24	26



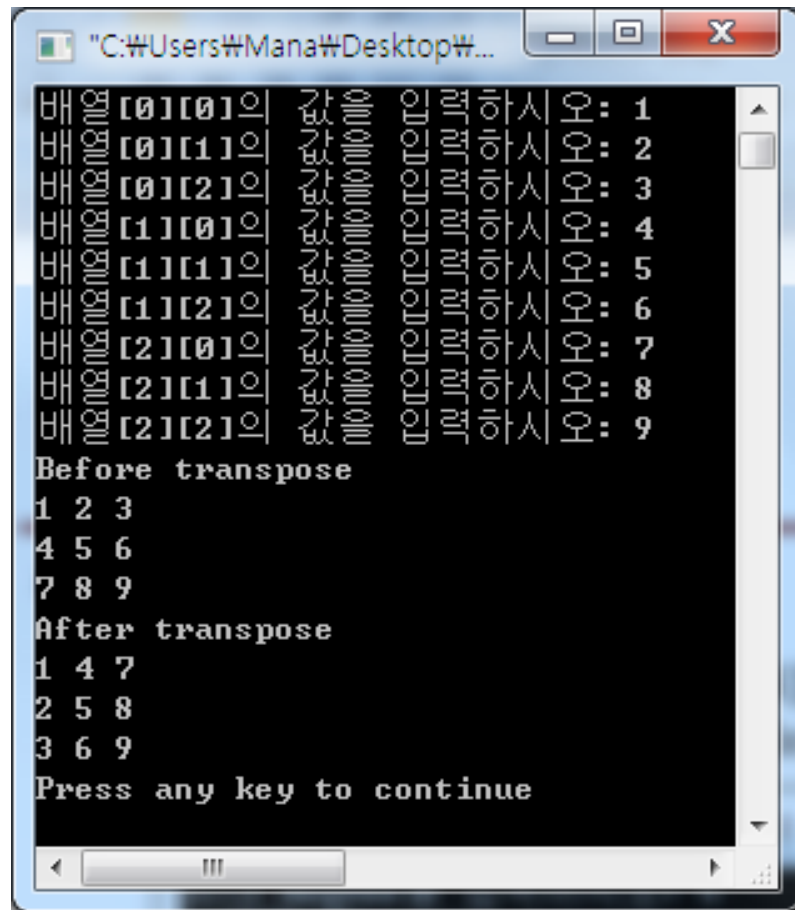
(0,0)	(0,1)	(0,2)
10	16	22
(1,0)	(1,1)	(1,2)
12	18	24
(2,0)	(2,1)	(2,2)
14	20	26

4. 배열 실습

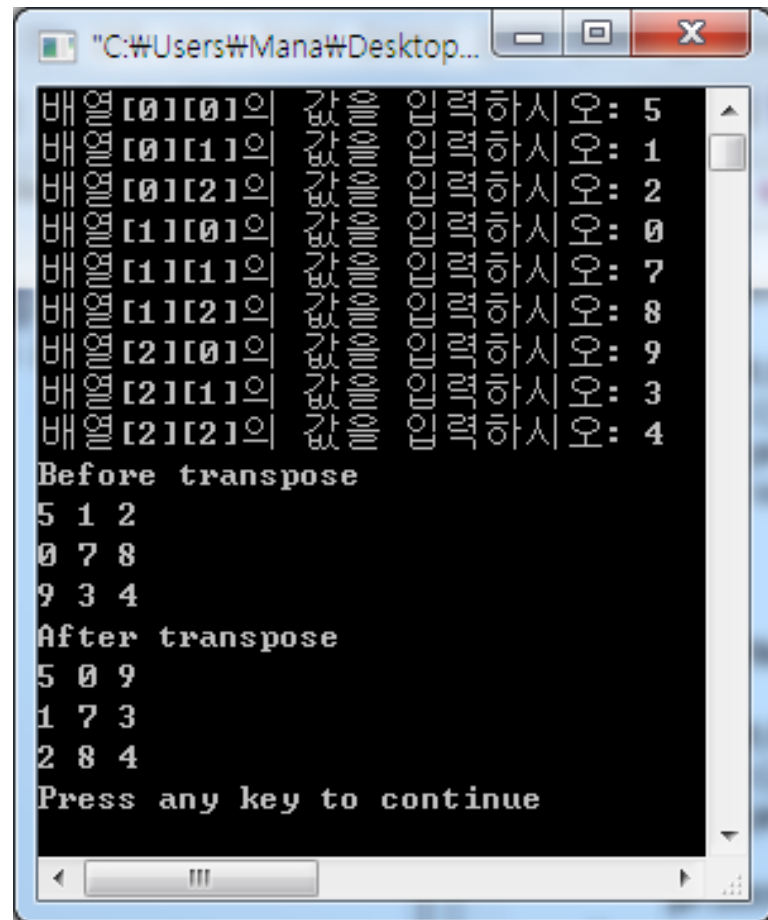
Contents

1. 포인터 실습
2. 성능 측정 실습
3. 팩토리얼 실습
4. 배열 실습

배열 실습하기 (2/2)



```
"C:\Users\Mana\Desktop\..."
배열 [0][0]의 값은 1이고 [0][1]의 값은 4이고 [0][2]의 값은 7이고 [1][0]의 값은 4이고 [1][1]의 값은 5이고 [1][2]의 값은 8이고 [2][0]의 값은 7이고 [2][1]의 값은 8이고 [2][2]의 값은 9이고
Before transpose
1 2 3
4 5 6
7 8 9
After transpose
1 4 7
2 5 8
3 6 9
Press any key to continue
```



```
"C:\Users\Mana\Desktop\..."
배열 [0][0]의 값은 5이고 [0][1]의 값은 1이고 [0][2]의 값은 0이고 [1][0]의 값은 9이고 [1][1]의 값은 3이고 [1][2]의 값은 4이고 [2][0]의 값은 5이고 [2][1]의 값은 0이고 [2][2]의 값은 9이고
Before transpose
5 1 2
0 7 8
9 3 4
After transpose
5 0 9
1 7 3
2 8 4
Press any key to continue
```

Q & A

