

Programming Methodology

Practice Session #2

Array and Recursion

Array

- 같은 **type**의 데이터 **여러 개를 묶어서** 저장하고 다루는 자료구조
- **정수형의 index**를 통해 **element**에 접근할 수 있다.

```
int intArray[5]; // array 선언
int i;
```

```
intArray[0] = 5; // element에 접근
intArray[1] = 1;
intArray[2] = 7;
intArray[3] = 8;
intArray[4] = 4;
```

index : 0 1 2 3 4

5	1	7	8	4
---	---	---	---	---

```
for(i=0; i<5; i++)
    printf("%d ", intArray[i]); // 출력
```

Recursion (1)

- 수행 도중에 자기 자신을 호출하도록 구현된 함수
- 점화식으로 나타낼 수 있는 순환적 구조를 가지는 문제는 recursion을 이용하면 코드를 직관적으로 이해하기 쉽다.
- 무한 반복의 위험이 있으므로 반드시 종료 조건을 설정해 주어야 한다.

Recursion (2)

```
int Factorial(int n)
{
    // 종료 조건, n == 2일 때 2를 return해도 됨.
    if(n == 1) return 1;

    // Recursion,  $n! = n * (n-1)!$ 
    return n * Factorial(n - 1);
}

void main()
{
    int result = Factorial(5);
    printf("5! = %d\n", result);    // 5! = 120
}
```

Sample Practice (1)

- **Recursion**을 이용하여 **Merge sort**를 구현한다.
 - `void mergeSort(int* arr, int size)`
- Merge sort는 **Divide and Conquer**의 두 단계로 이루어진다.
 - **Divide** : 주어진 `array`를 반으로 나누고, 나누어진 두 개의 `array`에 대해 각각 `mergeSort()` 함수를 재귀 호출한다. 이러한 재귀호출은 더 이상 `array`를 나눌 수 없을 때까지(`size == 1`, 종료 조건) 계속된다.
 - **Conquer** : 부분적으로 정렬된 두 개의 `array`를 하나로 합치면서 정렬한다.

Sample Practice (2)

- Merge Sort

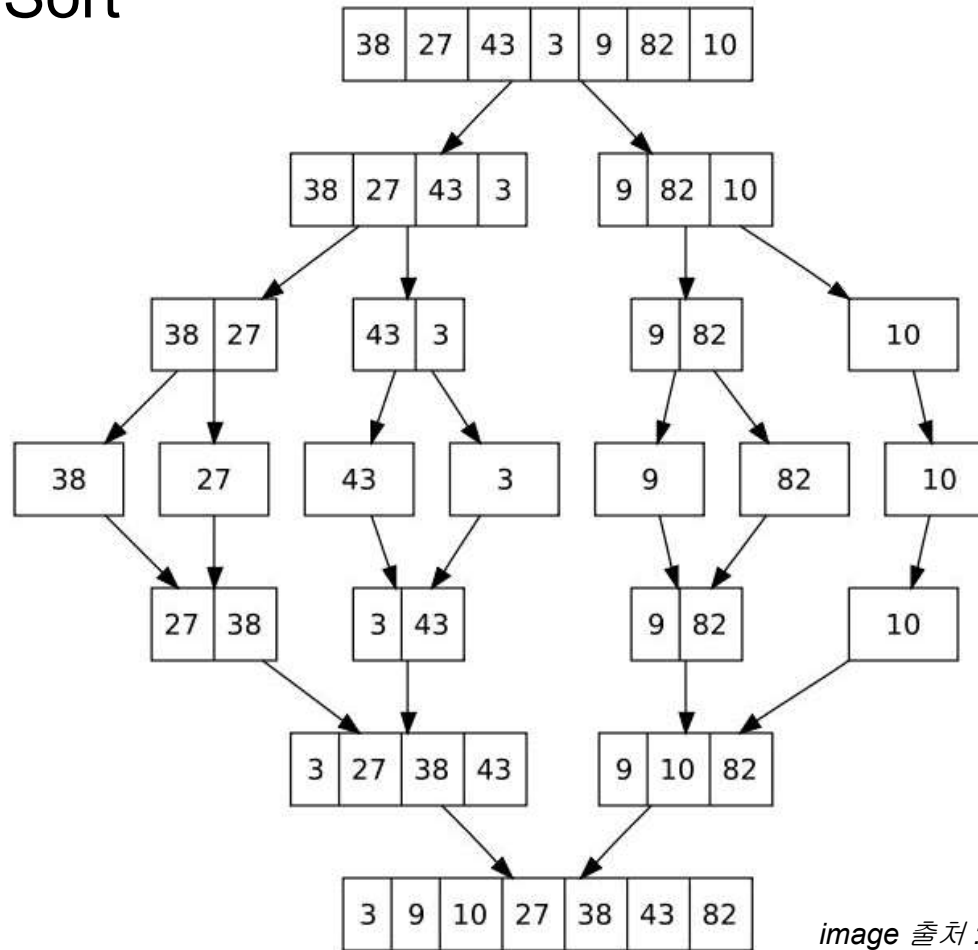
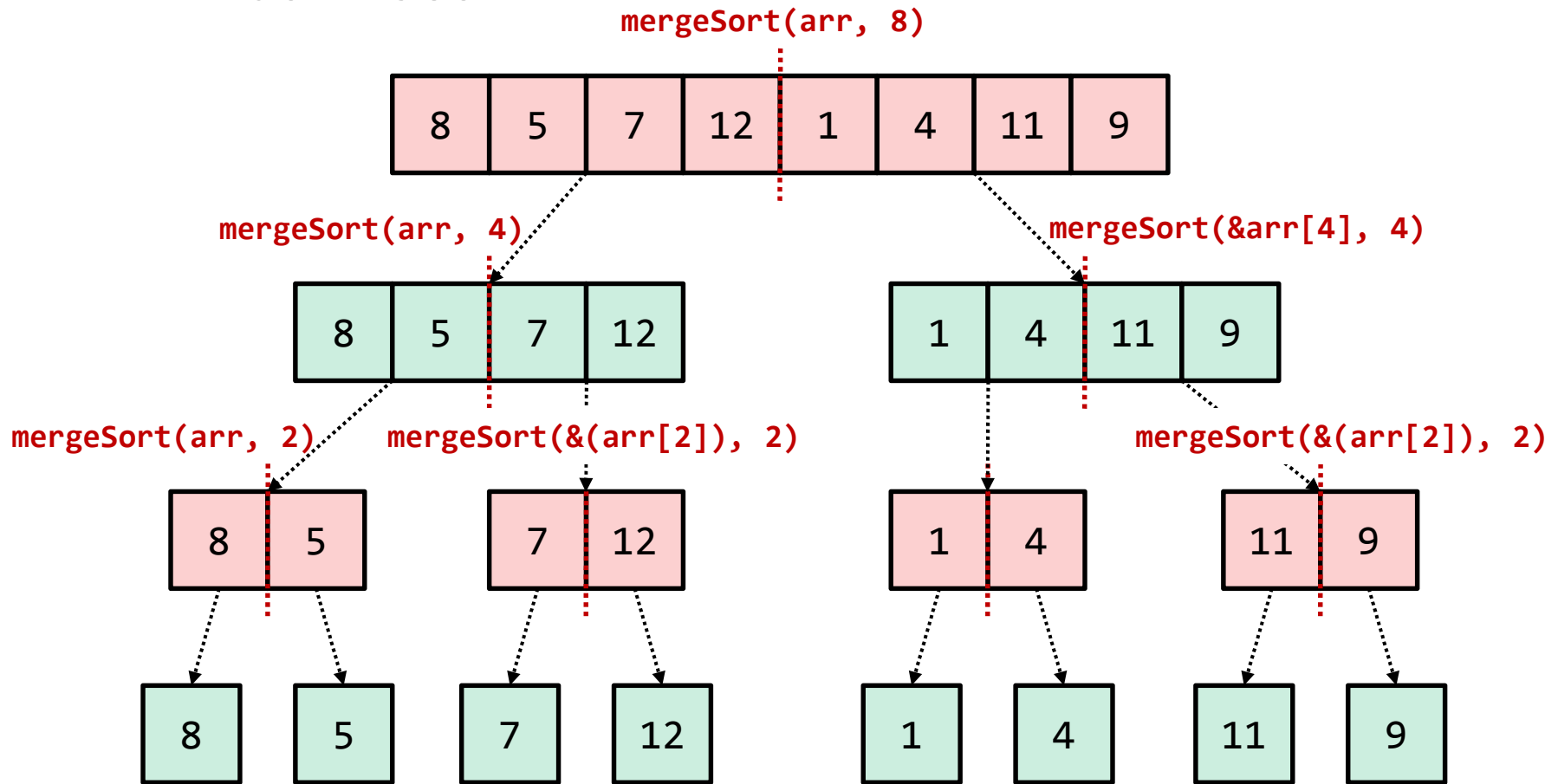


image 출처 : Wikipedia

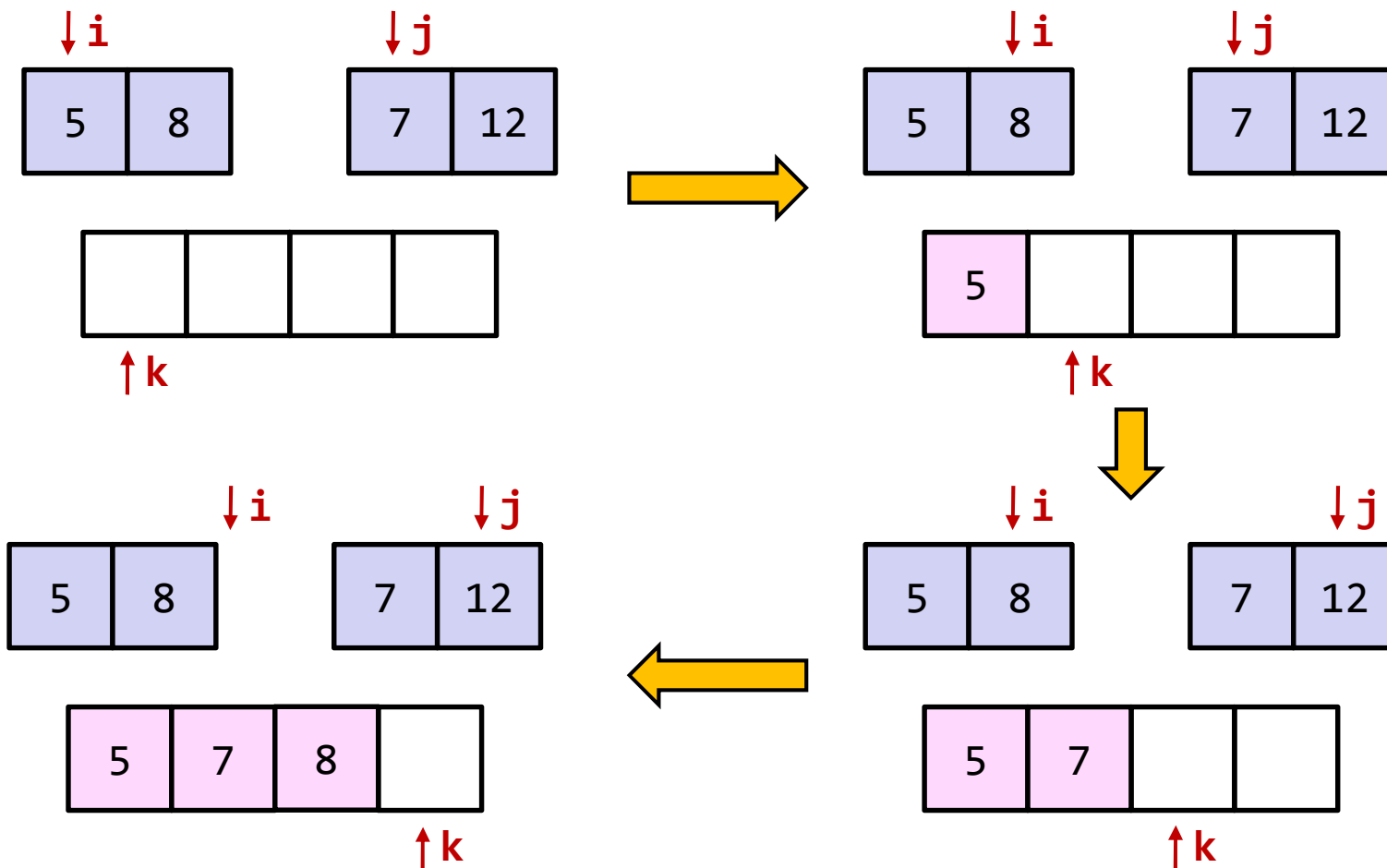
Sample Practice (3)

- Divide Phase



Sample Practice (4)

- Conquer Phase (Merging)



Sample Practice (5)

- 간단함을 위해 Array의 size는 **2의 거듭제곱 (2^n)** 인 것으로 가정한다.
- Merging할 때 **임시 array**를 선언하여 두 array를 합치고, 끝나면 원래의 array에 copy하는 것이 간편하다.

Sample Practice (6)

```
void main()
{
    int arr[16] = {8, 5, 7, 12, 1, 4, 11, 9,
                  2, 10, 3, 0, 13, 1, 4, 6};

    int i;

    printf("Before Merge Sort : ");
    for(i=0; i<16; i++)
        printf("%d ", arr[i]);
    printf("\n");

    mergeSort(arr, 16);

    printf(" After Merge Sort : ");
    for(i=0; i<16; i++)
        printf("%d ", arr[i]);
    printf("\n");
}
```

```
Before Merge Sort : 8 5 7 12 1 4 11 9 2 10 3 0 13 1 4 6
After Merge Sort : 0 1 1 2 3 4 4 5 6 7 8 9 10 11 12 13
계속하려면 아무 키나 누르십시오 . . .
```