



Lab 06

2023학년도 2학기 프로그래밍개론

01, 02, 03분반

숙명여자대학교 소프트웨어학부

데이터 지능 연구실

TA 유사라

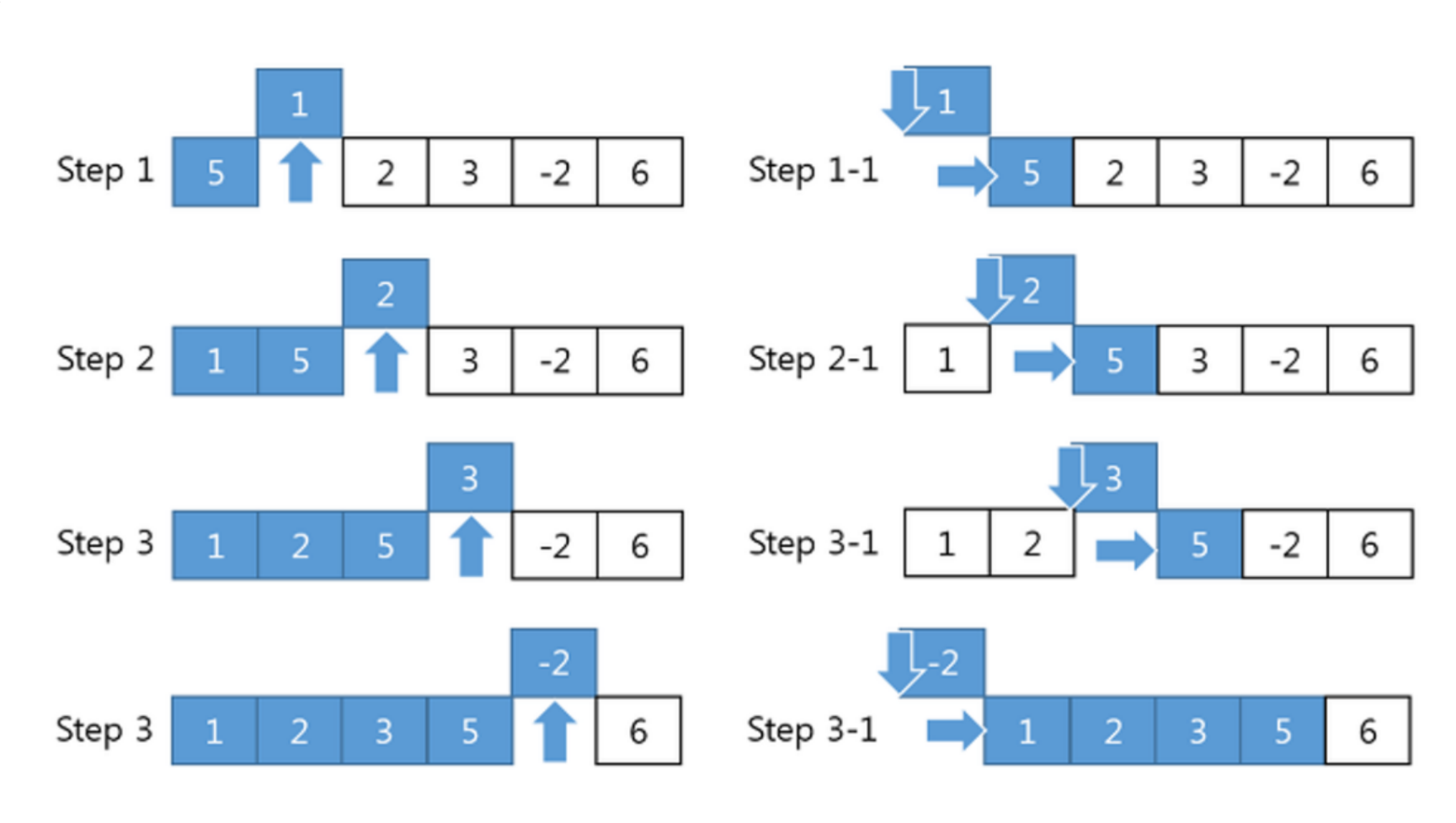
4ra@sookmyung.ac.kr

- 과제 1: 정렬 함수의 분석

- 정렬(sorting)
 - 일련의 항목들을 특정 순서로 나열하는 것
 - 어떤 값을 기준으로 정렬
 - 일련의 점수들을 작은 값에서 큰 값 순으로 정렬
 - 일련의 사람 이름들을 알파벳 순서로 정렬
 - 성능이 다른 많은 알고리즘이 있다.
- 정렬 알고리즘 예: 삽입 정렬 (Insertion Sort)

• 예 :

original : 5 1 2 3 -2 6



- 삽입 정렬 개요

첫 번째 항목을 한 항목을 갖는 정렬된 서브리스트로 생각한다.

두 번째 항목을 이 정렬된 서브리스트의 적절한 위치에 삽입한다.

세 번째 항목을 이 정렬된 서브리스트의 적절한 위치에 삽입한다.

모든 값들이 정렬 상태를 유지하며 적절한 위치에 삽입될 때까지 반복한다.

- 예 :

original: 3 9 6 1 2

insert 9: 3 9 6 1 2

insert 6: 3 6 9 1 2

insert 1: 1 3 6 9 2

insert 2: 1 2 3 6 9

- 삽입 정렬 함수 isort()

```
void isort(int v[], int n)
{
    int i, j, temp;
    for (i = 1; i < n; i++){
        for (j = i-1; j >= 0 && v[j] > v[j+1]; j--){
            temp = v[j], v[j] = v[j+1], v[j+1] = temp;
        }
    }
}
```

```
void isort(int v[], int n)
{
    int i, j, temp;
    for (i = 1; i < n; i++) { // Step 1, 2, 3, ... 의 비교할 수
        for (j = i-1; j >= 0 && v[j] > v[j+1]; j--) { // Step n-1 단계
            temp = v[j], v[j] = v[j+1], v[j+1] = temp;
        }
    }
} // 앞에 수들과 비교
```

• 예 :

Step 1 부분

바깥 for문 : index 1인 숫자 1 해당

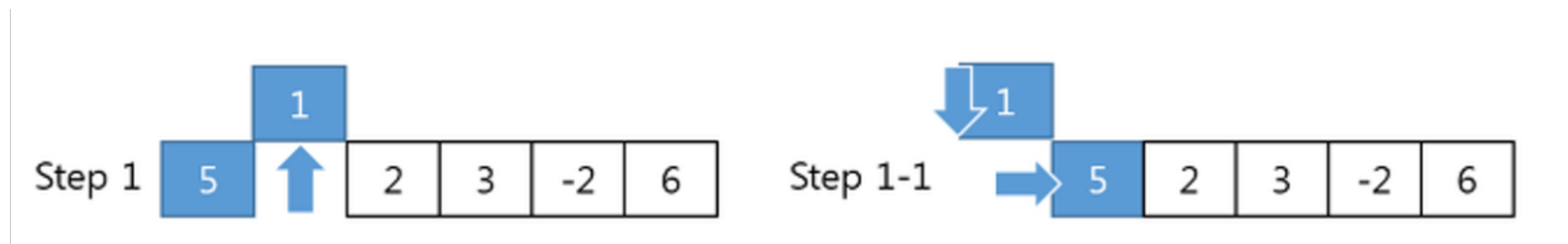
안쪽 for문 :

$j = i-1$ 이므로 index 0인 숫자 5

$j \geq 0$ // 현재 i앞에 비교할 수가 있고

&&

$5 > 1$ 이므로 두 수 위치 바꿈



과제 1: 정렬 함수의 분석(sort.c)

- 강의자료의 `isort()` 함수를 수정하여 **배열 원소의 교환 횟수**를 리턴하시오.
(`temp = v[j], v[j] = v[j+1], v[j+1] = temp`의 **수행 횟수**)
- 다음 페이지의 프로그램을 완성하기 위해 주어진 설명에 따라 **함수 `copyarray()`와 `printarray()`**도 함께 완성하시오.
- 이 프로그램의 실행 결과의 일부는 다음과 같다.

Before sort (the first 8 numbers) : 41 18467 6334 26500 19169 15724 11478 29358

After insertion sort (the first 5 numbers) : 41 6334 11478 15724 18467

The number of swaps in insertion sort (n = 10) : 13

After insertion sort (the first 5 numbers) : 41 153 288 292 491

The number of swaps in insertion sort (n = 100) : 2360

...

과제 1: 정렬 함수의 분석(sort.c)

- 강의자료의 `isort()` 함수를 수정하여 **배열 원소의 교환 횟수**를 리턴하시오.
(`temp = v[j]`, `v[j] = v[j+1]`, `v[j+1] = temp`의 **수행 횟수**)
- 다음 페이지의 프로그램을 완성하기 위해 주어진 설명에 따라 **함수 `copyarray()`**와 **`printarray()`**도 함께 완성하시오.
- 이 프로그램의 실행 결과의 일부는 다음과 같다.

Before sort (the first 8 numbers) : 41 18467 6334 26500 19169 15724 11478 29358

After insertion sort (the first 5 numbers) : 41 6334 11478 15724 18467

The number of swaps in insertion sort (n = 10) : 13

After insertion sort (the first 5 numbers) : 41 153 288 292 491

The number of swaps in insertion sort (n = 100) : 2360

...

41 ... : `printarray()`함수 사용 위치

13 ... : `isort()`함수 사용 위치

과제 1: 정렬 함수의 분석(sort.c)

numbers[10,000]에 10,000개의 랜덤한 수 담음 -> 앞의 8개만 출력

```
Before sort (the first 8 numbers) : 41 18467 6334 26500 19169 15724 11478 29358
```

data[]에 n(처음은 10개)를 복사 (copyarray() 함수 사용)

복사한 숫자가 담긴 data[] 삽입 정렬 (isort() 함수 사용)

isort() 함수 결과 정렬된 data[]배열의 앞의 5개만 출력

```
After insertion sort (the first 5 numbers) : 41 6334 11478 15724 18467
```

n개 복사한 data[] 삽입 정렬 결과 **// isort()함수에 count변수 추가하기**

isort() 함수에서 'temp = v[j], v[j] = v[j+1], v[j+1] = temp' 수행 횟수 출력

```
The number of swaps in insertion sort (n = 10) : 13
```

과제 1: 정렬 함수의 분석(sort.c)

- 출력 결과 다를 시 감점 (개행, 출력 결과 모두 꼼꼼히 보기)
- 출력 결과에서 계속 바뀌는 부분을 확인하면서 코드 작성 바람
 - 해당 부분은 변수가 출력하도록 printf문 작성시 유의

Before sort (the first 8 numbers) : 41 18467 6334 26500 19169 15724 11478 29358

(한 줄 공백)

After insertion sort (the first 5 numbers) : 41 6334 11478 15724 18467

The number of swaps in insertion sort (n = 10) : 13

(한 줄 공백)

After insertion sort (the first 5 numbers) : 41 153 288 292 491

The number of swaps in insertion sort (n = 100) : 2360

(한 줄 공백)

...

41 ... : 출력 결과 계속 바뀌는 부분

과제 1: 정렬 함수의 분석(sort.c)

삽입 정렬 수행 함수 -> 배열 원소의 교환 횟수 출력 필요

```
int isort(int v[], int n); /* insertion sort */
```

배열 b의 원소 n개 배열 a로 복사 하는 함수 -> 복사를 위한 반복문 필요

```
void copyarray(int a[], int b[], int n); /* copy n elements of array b[] to a[] */
```

배열 a의 원소 n개 출력하는 함수 -> 출력을 위한 반복문 필요
-> 숫자 하나 출력 후 띄어쓰기

```
void printarray(int a[], int n); /* print n elements of array a[] */
```

과제 1: 정렬 함수의 분석(sort.c)

```
#include <stdio.h>
#include <stdlib.h>
#define ARRAYSIZE 10000

int isort(int v[], int n){
    /* [과제] insertion sort 구현 */
}

void copyarray(int a[], int b[], int n){
    /* [과제] copy n elements of array b[] to a[] */
}

void printarray(int a[], int n){
    /* [과제] print n elements of array a[] */
}

// 다음 페이지에서 main() 이어짐
```

과제 1: 정렬 함수의 분석(sort.c)

```
main() {  
    int numbers[ARRAYSIZE], data[ARRAYSIZE];  
    int i, n, count;  
  
    /* Generate 10,000 random numbers */  
    for (i = 0; i < ARRAYSIZE; i++) {  
        numbers[i] = rand();  
    }  
    printf("Before sort (the first 8 numbers) : ");  
    printarray(numbers, 8);  
    printf("\n");  
    for (n = 10; n <= ARRAYSIZE; n *= 10) {  
        copyarray(data, numbers, n);  
        count = isort(data, n);  
        printf("After insertion sort (the first 5 numbers) : ");  
        printarray(data, 5);  
        printf("The number of swaps in insertion sort : %d\n\n", count);  
    }  
}
```

numbers[] : 모든 난수 담은 배열
data[] : numbers[] 중 n개
복사하여 정렬할 배열

numbers[]에 10000개 난수 생성
(rand())하여 담은 후,
그 중 앞 8개만 출력

n을 10, 100, 1000 , ... 늘려가며
data[]에 복사
data[] 삽입 정렬 후 결과 출력

- 제출 시 .c 파일 유의해야 할 점

1. main() 내용 수정, 삭제, 추가 금지
2. isort(), copyarray(),
printarray() 함수명 그대로 사용

Microsoft Visual Studio 디버그 콘솔

Before sort (the first 8 numbers) : 41 18467 6334 26500 19169 15724 11478 29358

After insertion sort (the first 5 numbers) : 41 6334 11478 15724 18467

The number of swaps in insertion sort : 13

After insertion sort (the first 5 numbers) : 41 153 288 292 491

The number of swaps in insertion sort : 2360

After insertion sort (the first 5 numbers) : 28 28 41 53 58

The number of swaps in insertion sort : 248145

After insertion sort (the first 5 numbers) : 3 4 8 9 12

The number of swaps in insertion sort : 24952888

- 제출할 때 다음과 같은 결과가 출력되면 완료

- 과제 제출 기한
 - 01, 02분반 : 10월 31일 (화) 23:59까지 스노보드로 제출
 - 03분반 : 11월 1일 (수) 23:59까지 스노보드로 제출
 - 중간고사 기간을 고려하여 1주 연장 => 기간 : 2주
 - 제출 기한 이후 24시간 이내에 지각 제출할 경우 : 2점 감점
 - 그 이후는 0점 처리(추가 제출 받지 않음)
- 제출 장소
 - 스노우보드 과제 제출 페이지에 업로드

- 소스파일(.c)과 과제보고서(.docx)가 담긴 **압축파일(.zip) 제출**
 - 압축 파일 이름: Lab06_학번_이름.zip
 - 'Lab06_학번_이름'으로 된 c파일 + 'Lab06_학번_이름'으로 된 .docx파일
 - 과제보고서(.docx) 양식
 - 스노우보드에서 다운로드
1. 실행 결과 화면 캡처한 이미지 첨부
 2. 소스 코드
 3. 소스 코드에 대한 설명 (간략하게 3-4줄)

- 조교 메일로 질문 보내기
 - 4ra@sookmyung.ac.kr

- 질문시 주의사항

- 메일에 반드시 과목, 분반, 전공, 이름, 학번 명시

제목 : [프로그래밍개론 001분반] 2331297 유사라 Lab01 질문 드립니다.
- 몇 번 과제에서 어떤 부분이 막혔는지, 어떤 부분이 문제인지 코드와 함께 설명 첨부
(그냥 코드만 보내면 어디가 문제인지 알 수 없어요)
- 답장이 늦을수도 있으니 이 점 고려하여 미리 질문 (특히 과제 제출 마지막날 유의!)
- 질문 내용을 구체적으로 명확하게 적어 주시기 바랍니다.
- 오류 메시지를 첨부하고 싶을 경우, 오류 캡처 화면 + 전체 코드 c 파일을 첨부하여 보내주세요. (코드 캡처 사진 X)
- 그 외 출석 등 다른 질문들도 메일로