

# 실습9 - Sorting



## 실습9 정렬

# 실습 목적 및 정렬(sorting)의 이해

## ■ 실습 목적

- 정렬 방법들을 이해한다.
- 정렬 알고리즘들을 구현해보고, 각 정렬 알고리즘의 효율을 확인해 본다.
- 모든 경우에 대해 최적인 정렬 알고리즘은 없음
  - 해당 응용 분야에 적합한 정렬 방법 사용해야 함
- Key의 특성(문자, 정수, 실수 등)
- 메모리 내부/외부 정렬
- 단순하지만 비효율적인 방법 : 삽입, 선택, 버블정렬 등
- 복잡하지만 효율적인 방법: 퀵, 힙, 합병, 기수정렬 등

# 실습 목적 및 정렬(sorting)의 이해

## ■ 선택정렬



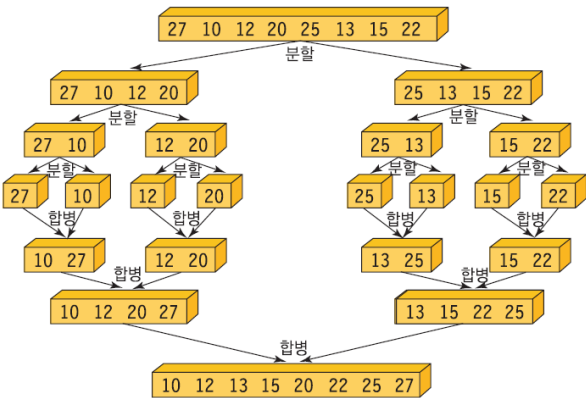
## ■ 삽입정렬



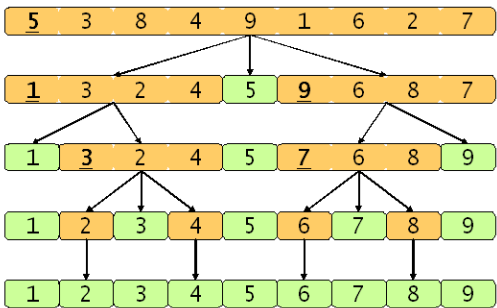
## ■ 버블정렬



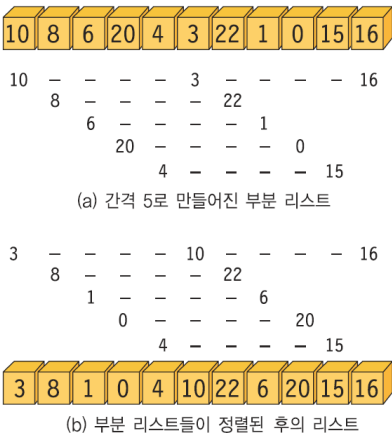
## ■ 합병정렬



## ■ 퀵정렬



## ■ 쉘정렬



# 실습 문제 - 정렬

## ■ 실습 프로그램 설명

- 주어진 배열을 정렬한 뒤, 정렬 과정과 정렬 결과를 출력해본다.
- 다른 정렬 방식과 정렬 방법을 비교한다.

## ■ 주어진 함수

- 정렬 함수
  - void selection\_sort(int list[], int n)
  - void insertion\_sort(int list[], int n)
  - void insertion\_sort\_fn(int list[], int n, int(\*f)(int, int))
  - void bubble\_sort(int list[], int n)
  - void shell\_sort(int list[], int n)
  - static void sortGapInsertion(int list[], int first, int last, int gap)
  - static void merge(int list[], int left, int mid, int right)
  - void merge\_sort(int list[], int left, int right)
  - int partition(int list[], int left, int right)
  - void quick\_sort(int list[], int left, int right)

## ■ 기타 함수

- void printStep(int arr[], int n, int val)
  - 정렬의 중간 과정을 출력하는 함수
- void printArray(int arr[], int n, char \*str)
  - 배열의 내용을 순서대로 출력하는 함수
- 정렬에 대한 설명은 Sorting 강의노트 참조

# 실습 문제 - 정렬

## ■ 실습 문제 : 정렬의 구현 및 사용

- 정수형 배열을 정렬하는 다양한 정렬 알고리즘을 구현해 본다.
- 정렬의 방법을 다른 정렬 방식과 비교한다.

```
void main()
{
    int sort_method = 0;
    char end[5]="n";

    printArray(list, size, "Original ");    // 정렬 전 배열 출력
    do{

        list[0] = 5, list[1]= 3, list[2]=8, list[3]=4, list[4]=9, list[5]=1, list[6]=6, list[7]=2, list[8]=7;
        printf("\n\n");
        printArray(list, size, "Original ");    // 정렬 전 배열 출력
        printf("(1번: 선택정렬, 2번: 삽입정렬, 3번: 버블정렬, 4번: 내림차순 삽입정렬, 5번: 셸 정렬, 6번: 합병 정렬, 7번: 퀵 정렬, (1-7)이외의 숫자 번호: 종료)\n 정렬 방법을 선택하세요: ");
        scanf("%d", &sort_method);
        switch(sort_method){
            case 1:
                selection_sort(list, size);    // 선택 정렬 실행
                printArray(list, size, "Selection");    // 정렬 후 배열 출력
                break;
            case 2:
                insertion_sort(list, size);    // 삽입 정렬 실행
                printArray(list, size, "Insertion");    // 정렬 후 배열 출력
                break;
            case 3:
                bubble_sort(list, size);    // 버블 정렬 실행
                printArray(list, size, "Bubble ");    // 정렬 후 배열 출력
                break;
            case 4:
                insertion_sort_fn(list, size, descend);    // 내림차순 삽입정렬 알고리즘
                printArray(list, size, "Insert-De");    // 정렬 후 배열 출력
                break;
            case 5:
                shell_sort(list, size);    // 셸정렬 알고리즘
                printArray(list, size, "ShellSort");    // 정렬 후 배열 출력
                break;
            case 6:
                merge_sort(list, 0, size-1);    // 합병정렬 알고리즘
                printArray(list, size, "MergeSort");    // 정렬 후 배열 출력
                break;
            case 7:
                //quick_sort(list, 0, size-1);    // 퀵정렬 알고리즘
                qsort((void *)list, size, sizeof(int), compare);    // 퀵정렬 라이브러리 함수 호출
                printArray(list, size, "QuickSort");    // 정렬 후 배열 출력
                break;
            default:
                printf("정렬을 끝내시겠습니까?(y/n) :");
                scanf("%s",end);
                break;
        }
    } while (strcmp(end,"n")!=0);
}
```

# 실습 문제 - 정렬

- 실습 문제 : 정렬의 구현 및 사용
- 메인 함수 설명
  - 각 정렬에 대하여, 정렬 방법 선택 -> 정렬 -> 정렬 확인을 반복
  - 각 정렬 함수의 수행 과정을 확인해볼 수 있다.
  - 종료 조건을 입력하면 정렬을 종료

## ■ 결과 화면

```
C:\Windows\system32\cmd.exe
Original = 5 3 8 4 9 1 6 2 7
Original = 5 3 8 4 9 1 6 2 7
<1번: 선택정렬, 2번: 삽입정렬, 3번: 버블정렬, 4번: 내림차순 삽입정렬, 5번: 셸 정렬, 6번: 합병 정렬, 7번: 퀵 정렬, <1-7>이외의 숫자 번호: 종료>
정렬 방법을 선택하세요: 1
Step 1 = 1 3 8 4 9 5 6 2 7
Step 2 = 1 2 8 4 9 5 6 3 7
Step 3 = 1 2 3 4 9 5 6 8 7
Step 4 = 1 2 3 4 9 5 6 8 7
Step 5 = 1 2 3 4 5 9 6 8 7
Step 6 = 1 2 3 4 5 6 9 8 7
Step 7 = 1 2 3 4 5 6 7 8 9
Step 8 = 1 2 3 4 5 6 7 8 9
Selection = 1 2 3 4 5 6 7 8 9
```

```
C:\Windows\system32\cmd.exe
Original = 5 3 8 4 9 1 6 2 7
<1번: 선택정렬, 2번: 삽입정렬, 3번: 버블정렬, 4번: 내림차순 삽입정렬, 5번: 셸 정렬, 6번: 합병 정렬, 7번: 퀵 정렬, <1-7>이외의 숫자 번호: 종료>
정렬 방법을 선택하세요: 9
정렬을 끝내시겠습니까?<y/n> :n

Original = 5 3 8 4 9 1 6 2 7
<1번: 선택정렬, 2번: 삽입정렬, 3번: 버블정렬, 4번: 내림차순 삽입정렬, 5번: 셸 정렬, 6번: 합병 정렬, 7번: 퀵 정렬, <1-7>이외의 숫자 번호: 종료>
정렬 방법을 선택하세요: 0
정렬을 끝내시겠습니까?<y/n> :y
계속하려면 아무 키나 누르십시오 . . .
```

Thank You  
Q&A