

homework 2 COMP319 Algorithms Fall 2019

강길진

① Insertion Sort를 linked list를

이용하여 구현한다.

(array를 이용한 것은 주어진다)

방법 1) head ☒
tail ☒

head \rightarrow [2] \rightarrow ||
tail \nearrow

[2] \rightarrow [3] \rightarrow ||
 \uparrow head \uparrow tail

[2] \rightarrow [3] \rightarrow [4] \rightarrow ||
 \uparrow head \uparrow tail

~~sorted~~ unsorted [5] \rightarrow [4] \rightarrow [6] \rightarrow [3] \leftarrow [2]
find-min

unsorted [5] \rightarrow [4] \rightarrow [6] \rightarrow [3] \rightarrow ||
find-min
tail 주기에 붙인다

[5] \rightarrow [4] \rightarrow [6] \rightarrow ||
find-min

[5] \rightarrow [6] \rightarrow ||

방법 2) head \rightarrow ||

head \rightarrow [6] \rightarrow ||

head \rightarrow [5] \rightarrow [6] \rightarrow ||

[4] \rightarrow [5] \rightarrow [6] \rightarrow ||
 \uparrow head

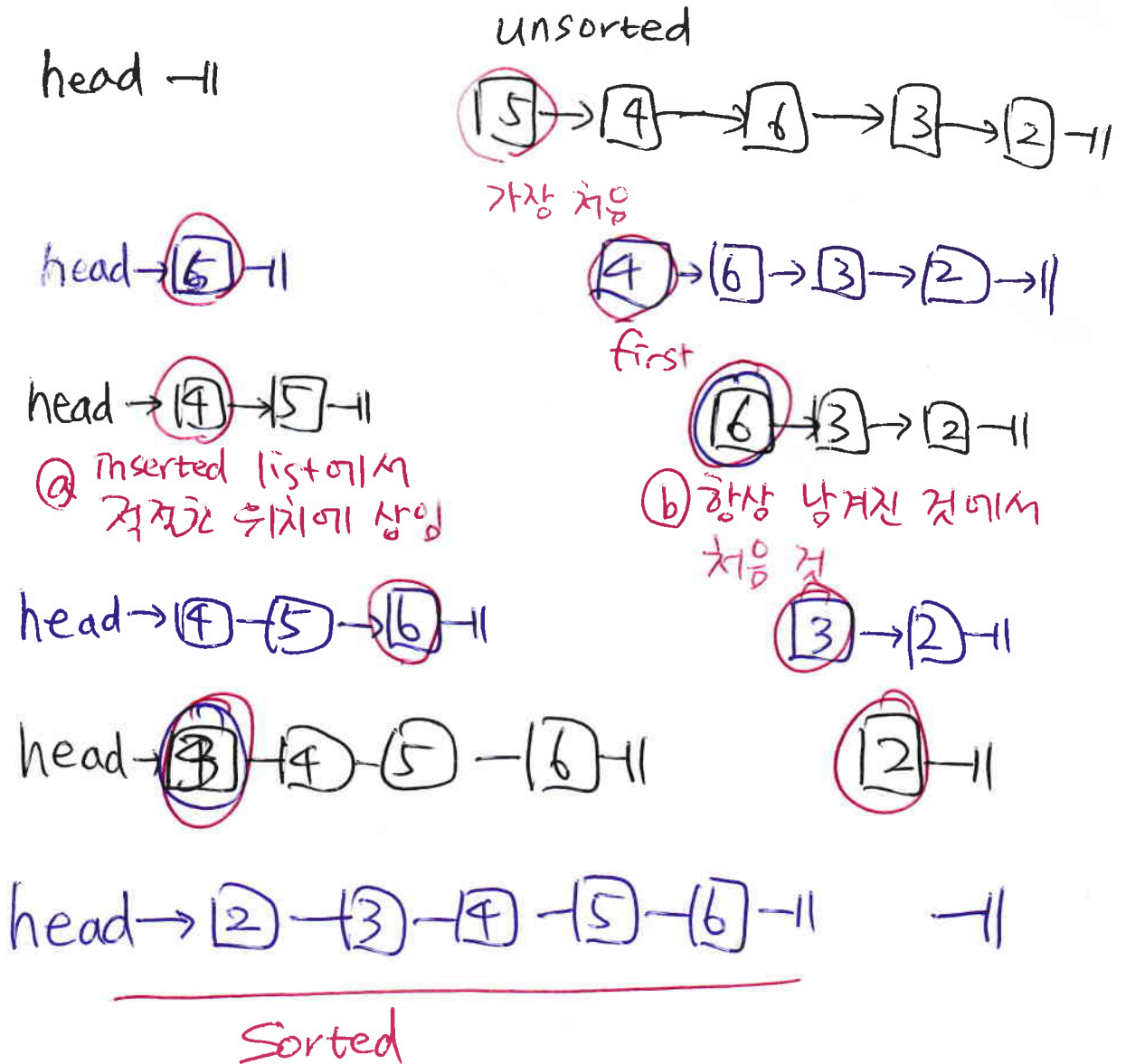
unsorted [5] \rightarrow [4] \rightarrow [6] \rightarrow [3] \rightarrow [2] \rightarrow ||
find-max

[5] \rightarrow [4] \rightarrow [3] \rightarrow [2] \rightarrow ||
find-max
head 앞에 붙인다

[4] \rightarrow [3] \rightarrow [2] \rightarrow ||
find max

[3] \rightarrow [2] \rightarrow ||

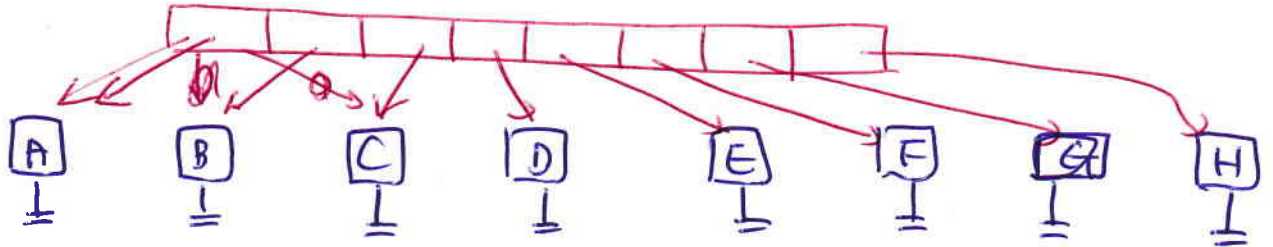
방법 3) 방법 1과 방법 2는 강의자료에 있는 Insertion Sort와 다르다. Selection Sort에서 Swap 대신 linked list에 넣는 변형된 Selection Sort이다.
강의자료의 Insertion Sort와 동일한 방법은 다음과 같다.



② merge sort 를 array 를 이용하여 구현한다.

③ merge sort 를 list 를 이용하여 구현한다.

아이디어)



증거에는 독립된 list

pointer array가 필요

방법 1)

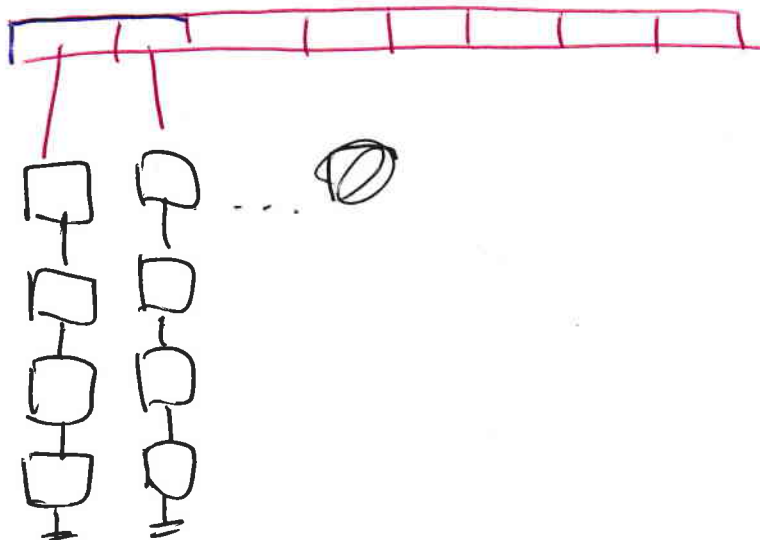
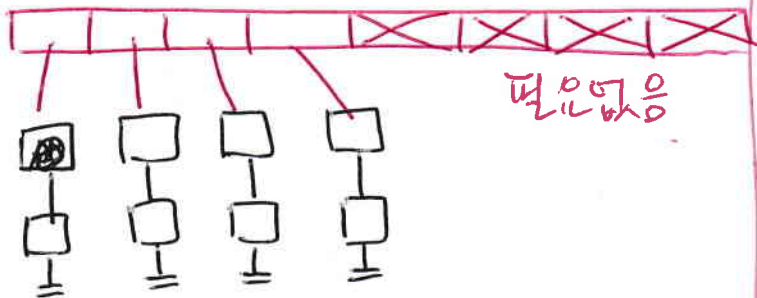
~~A B~~
merge

(A E) → A

(B F) → B

(C G) → C

(D H) → D



방법 2)

merge (A H) → A

(B G) → B

(C F) → C

(D E) → D

④ time challenge.