

REPORT



과목명 :	자료구조
담당 :	장형수 교수님
제출마감일 :	2021년 11월 15일
학과 :	생명과학과
학번 :	20171483
이름 :	조주현



SOGANG UNIVERSITY



SOGANG
UNIVERSITY

1. Problem description

<HW3-1>

Linked list의 길이 n과 node에 저장될 데이터(정수형 배열)를 입력받고, 정수 k를 입력받는다. List를 k만큼 오른쪽으로 밀고 난 뒤 list를 출력한다.

<HW3-2>

인쇄 대기목록을 받고, 우선순위가 높은 것부터 인쇄를 진행한다. 맨 앞부터 우선순위를 확인해 이 보다 높은게 뒤에 있다면 대기열의 가장 뒤로 보내고, 맨 앞에 것이 우선순위가 가장 높다면 출력을 진행한다. 인쇄한 목록과 인쇄 대기중인 목록을 매 인쇄시마다 출력한다.

<HW3-3>

웹 브라우저에서 주소 이동에 대한 함수들을 구현한다. 명령어는 Algorithm description과 함께 서술한다. 이 때 주소 이동에 대한 함수들은 클래스를 이용해 구현한다. 가장 초기 홈페이지의 주소는 <http://cs.sogang.ac.kr>이다.

2. Algorithm description

<HW3-1>

길이 n, 입력받은 정수열에 해당하는 list를 구현한다. 맨 마지막 노드를 노드 포인터를 이용해 찾고 첫 노드로 바꾼다. 이 함수를 k번 호출하면 k번 오른쪽으로 민 것과 동일하다.

<HW3-2>

인쇄 대기목록은 정수형 배열로 받았다. Linked queue로 배열을 바꿔준 뒤, linked queue의 가장 앞 노드의 data를 뒤에 있는 node들과 비교해 더 큰 값이 있으면 queue의 rear 포인터를 이용해 가장 뒤로 보낸다. 만약 큰 값이 없다면 해당 노드의 데이터를 이용해 새로운 linked queue를 생성하고, 원래의 linked queue에서 가장 앞 노드를 제거한다. 이후 원래의 linked queue, 새로 만든 linked queue를 모두 출력하고 원래의 linked queue의 가장 앞 노드의 크기를 비교하는 것으로 돌아가 반복한다.

<HW3-3>

기본적으로 현재주소, home 주소, favorite 주소의 linked list pointer를 갖는 node를 선언하고, 각각 next, prev 포인터로 앞 뒤 노드를 연결해준다. 다만 헤더노드와 마지막 노드를 연결하지 않는다. 아래의 menu에 해당하는 단어들을 구조체로 선언해주고 입력받은 menu와 비교해준다. 이 때 go addr, Change_home addr, add_favorites addr에서 처럼 주소는 menu와 함께 받는다. current_page는 현재 페이지를, header_page는 header_node를 가르킨다.

go addr : addr에 해당하는 주소로 이동한다. 새로운 node를 할당해 addr의 주소를 저장한 뒤 current_page와 연결한다.

forward : 다음 주소로 이동하고, 그 주소를 출력한다. 다만 다음 주소가 없다면 현재 주소를 출력한다. current_page의 다음 노드가 없다면 현재주소를, 다음 노드가 있다면 이동 후 그 노드의 주소를 출력한다.

backward : 이전 주소로 이동하고, 그 주소를 출력한다. 다만 이전 주소가 없다면 현재 주소를 출력한다. current_page의 이전 노드가 없다면 현재주소를, 이전 노드가 있다면 이동 후 그 노드의 주소를 출력한다.

Home : Home 주소로 이동한 뒤 주소를 출력한다. 새로운 node를 할당한 뒤, home 주소를 현재 주소 값에 저장해 current_page와 연결하고, current_page가 새로운 노드를 가르키게 한다.

Change_home addr : home 주소를 바꾼다. header_page부터 current_page home 주소를 바꾼다.

history : 지금까지 페이지 이동 기록을 출력한다. header_page부터 current_page까지 주소를 출력한다.

add_favorites addr: header_page의 favorite list pointer에 favorite 주소를 저장한 뒤 연결한다.

list_favorites : header_page의 favorite list를 출력한다.

quit : 프로그램 종료. page_node와 favorite 주소의 linked list에 할당된 메모리를 해제한뒤 프로그램을 종료한다.

3. Program output

HW3-1

```
n: 5
1 2 3 4 5
k: 2
4 5 1 2 3
Program ended with exit code: 0
```

```
n: 3
0 1 2
k: 4
2 0 1
Program ended with exit code: 0
```

```
n: 6
5 6 7 1 2 3
k: 3
1 2 3 5 6 7
Program ended with exit code: 0
```

HW3-2

```
9 / 8 2 1 5 3
9 8 / 2 1 5 3
9 8 5 / 3 2 1
9 8 5 3 / 2 1
9 8 5 3 2 / 1
9 8 5 3 2 1 /
Program ended with exit code: 0
```

HW3-3

```
http://cs.sogang.ac.kr
go naver.com
naver.com
go daum.com
daum.com
forward
Can't forward
```

backward

```
backward
http://cs.sogang.ac.kr
backward
Can't backward
http://cs.sogang.ac.kr
forward
```

forward

```
forward
Can't forward
```

```
history
http://cs.sogang.ac.kr
```

* current_page에 주소가 저장되지 않았지만

```
http://cs.sogang.ac.kr
go naver.com
naver.com
Home
http://cs.sogang.ac.kr
Change_home google.com
Home
```

공간은 할당되어있지만 저장되지 않았습니다.

```
http://cs.sogang.ac.kr
add_favorites naver.com
add_favorites daum.net
list_favorites
```

공간은 할당되어 있지만 저장되지 않았습니다.

```
http://cs.sogang.ac.kr
go naver.com
naver.com
go google.com
google.com
quit
Program ended with exit code: 0
```