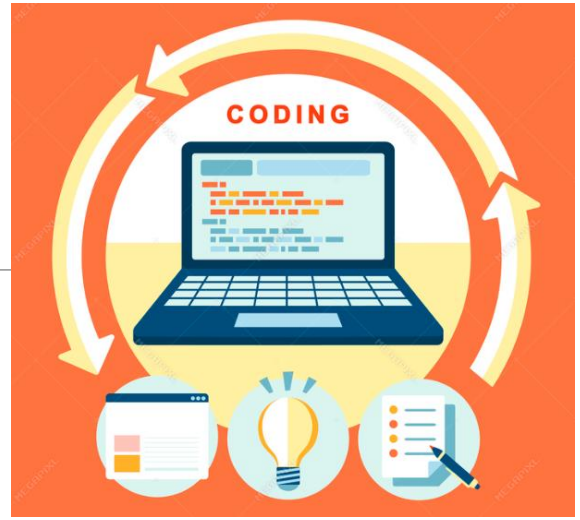
성균관대학교
SUNG KYUN KWAN UNIVERSITY(SKKU)

자 료 구 조

실 습 활 동



연결 리스트 (Linked List)

ADT(Abstract Data Type)

작업: ADT (구현자 관점)	명령어 (사용자 관점)	실행 결과 (자료 관점)
create(my_list)	ListNode* head;	
addTail(my_list, data)	+a +b +c +d	a b c **d**
print(my_list)	L	a b c **d**
get_data(my_list)	G	d (a b c **d**)
traverse_front(my_list, count)	< N N	a b **c** d

작업: ADT (구현자 관점)	명령어 (사용자 관점)	실행 결과 (자료 관점)
delete(my_list)	-	a b **d**
add(my_list, N, data) // N=current position + 1	+b +c +d	a b d b c **d**
traverse_front(my_list, count)	< N	a **b** d b c d
add(my_list, N, data)	+e +f +g	a b e f **g** d b c d
traverse_rear(my_list, count) delete(data)	> -	**a** b e f g d b c
add(my_list, 0, data)	< +x +y +z	x y **z** a b e f g d b c

replace(my_list, new_data)	=m	x y **m** a b e f g d b c
traverse_front(my_list, 5-1) get_data(my_list)	5G	b
data_count(my_list)	#	11
is_member(my_list, data)	?e	6: x y m a b **e** f g d b c
traverse_front(my_list, 2-1) replace(my_list, data)	2=z	x **z** m a b e f g d b c
// clear the list (i.e. for # {-})	C	
is_empty(my_list)	E	True

자신만의 기능을 3개 추가해보세요!