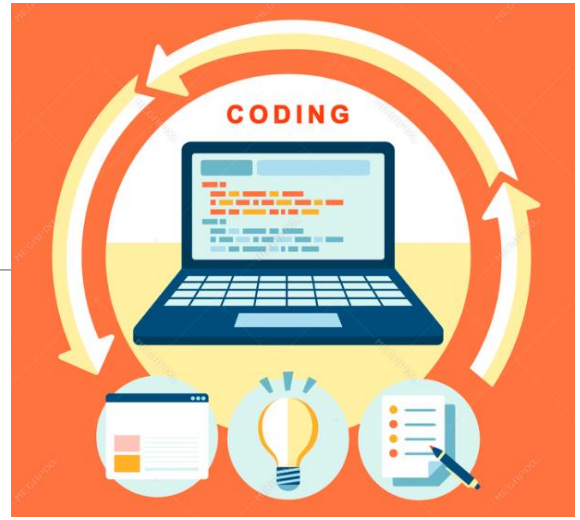
성균관대학교
SUNG KYUN KWAN UNIVERSITY(SKKU)

자 료 구 조

실 습 활 동



이진 탐색 트리

(Binary Search Tree)

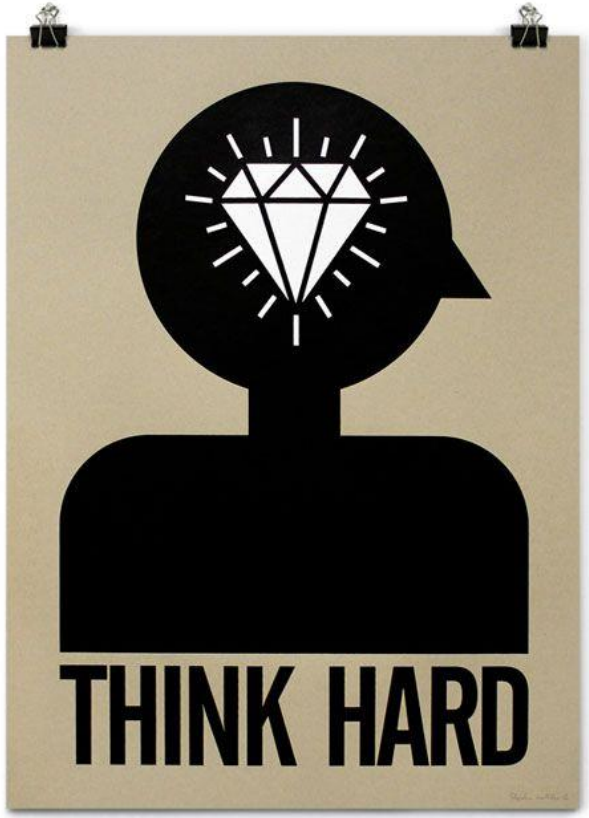
ADT(Abstract Data Type)

작업: ADT (구현자 관점)	명령어 (사용자 관점)	실행 결과 (자료 관점)
create(my_BST)	+33	(33)
insert_node(my_BST, new_node)	+22	(33(22,))
insert_node(my_BST, new_node)	+44	(33(22,44))
insert_node(my_BST, new_node)	+30	(33(22(,30),44))
insert_node(my_BST, new_node)	+40	(33(22(,30),44(40,)))
print(my_BST)	P	(33(22(,30),44(40,)))

작업: ADT (구현자 관점)	명령어 (사용자 관점)	실행 결과 (자료 관점)
inorder_traversal(my_BST)	I	22 30 33 40 44
right_root_left_traversal(my_BST)	R	44 40 33 30 22
get_min(my_BST)	N	22
get_max(my_BST)	X	44
find_node(my_BST, node_value)	F25	Error // Not Exist!
find_node(my_BST, node_value)	F40	Root > Right > Left
delete_node(my_BST, node)	-33	(30(22,44(40,))) [적용]
		(40(22(,30),44)

작업: ADT (구현자 관점)	명령어 (사용자 관점)	실행 결과 (자료 관점)
insert_node(my_BST, new_node)	+33	(30(22,44(40(33,),)))
height(my_BST)	H	3 [0부터 적용]
get_right_child(my_BST, node)	G(30)	44
get_left_child(my_BST, node)	L(22)	NULL
count_node(my_BST)	#	5
clear(my_BST)	C	

자신만의 기능을 3개 추가해보세요!



【도전 프로그램】

- 수치 자료를 입력 받아 max heap을 구성하는 프로그램을 작성한 후,

  max heap에서 binary search tree로 전환하는 프로그램을 구현해보세요.