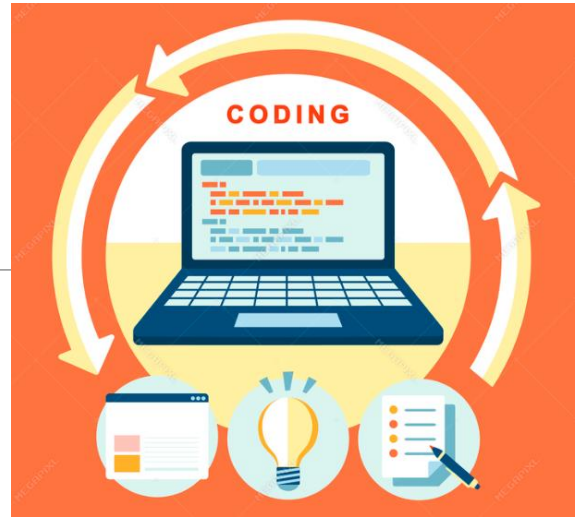
성균관대학교
SUNG KYUN KWAN UNIVERSITY(SKKU)

자 료 구 조

실 습 활 동



트리 (Tree)

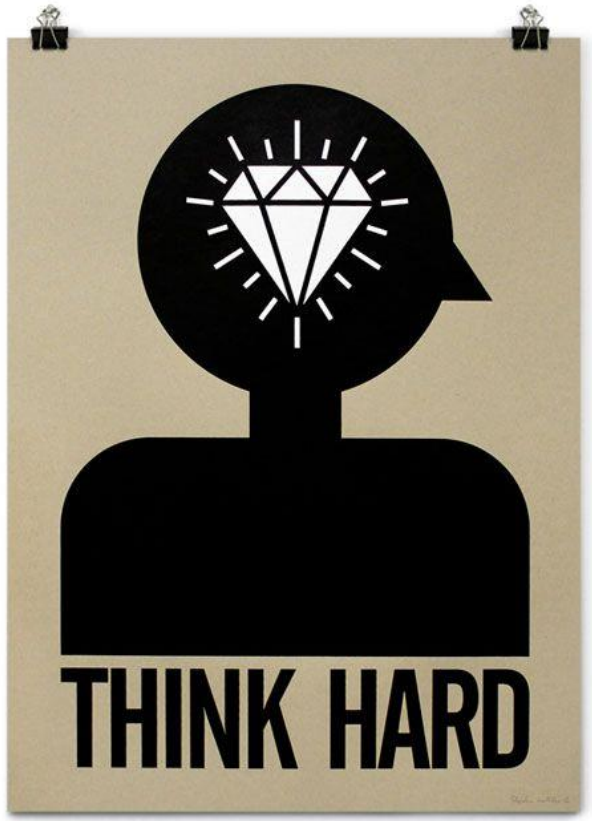
ADT(Abstract Data Type)

작업: ADT (구현자 관점)	명령어 (사용자 관점)	실행 결과 (자료 관점)
create(my_tree)	+^A	(A)
insert_node(my_tree, p_node, c_list)	+A(B,C,D)	(A(B,C,D))
insert_node(my_tree, p_node, c_list)	+A(E)	(A(B,C,D,E))
get_sibling(my_tree, node)	S(D)	{B,C,E}
insert_node(my_tree, p_node, c_list)	+F(G)	Error // No F
insert_node(my_tree, p_node, c_list)	+B(F,G)	(A(B(F,G),C,D,E))

작업: ADT (구현자 관점)	명령어 (사용자 관점)	실행 결과 (자료 관점)
print(my_tree)	T	(A(B(F,G),C,D,E))
get_parent(my_tree, node)	P(E)	A
get_child(my_tree, node)	C(A)	{B,C,D,E}
level_of_node(my_tree, node)	L(G)	2
level_of_tree(my_tree)	L	2
delete_node(my_tree, node)	-B	Error // parent node
delete_node(my_tree, node)	-G	(A(B(F),C,D,E))
insert_node(my_tree, p_node, c_list)	+F(H)	(A(B(F(H)),C,D,E))

작업: ADT (구현자 관점)	명령어 (사용자 관점)	실행 결과 (자료 관점)
get_ancestors(my_tree, data)	A(H)	F B A (A(B(F(H)),C,D,E))
get_descendants(my_tree, data)	D(B)	F H
degree_of_node(my_tree, node)	G(B)	1
degree_of_tree(my_tree)	G	4
count_node(my_tree)	#	7
insert_sibling(my_tree, node, s_list)	=+F(I,J)	(A(B(F(H),I,J),C,D,E))
join_trees(new_root, tree1, tree2)	J(P,my_tree,t)	(P((A(B(F(H),I,J),C,D,E)), -))
clear(my_tree)	K	

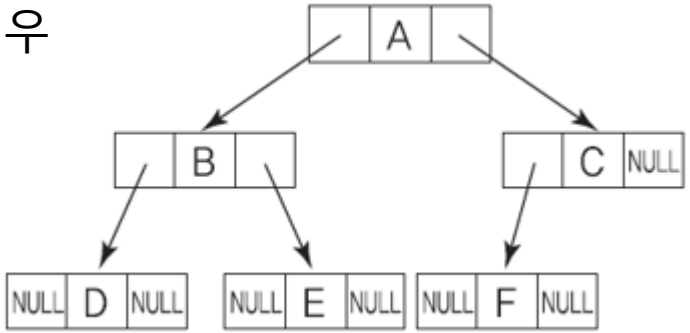
이진 트리를 위한 ADT를 추가해주세요!



프로그램이 실행되면

일반 트리 또는 이진 트리 선택하여 실행

이진 트리의 경우



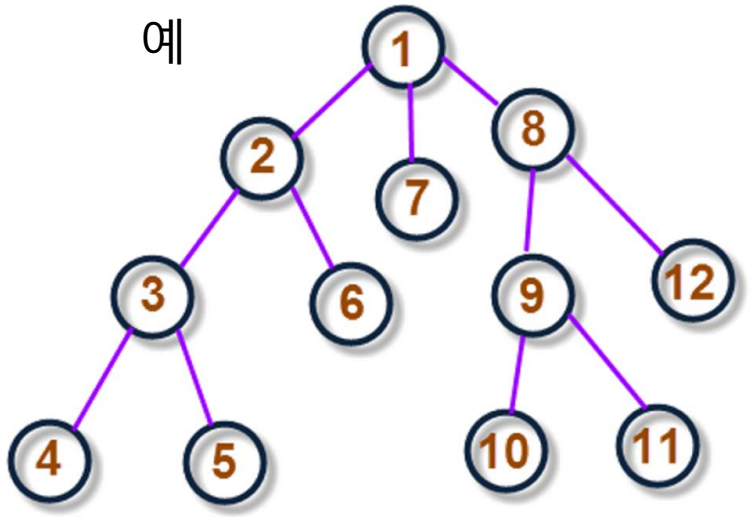
【도전 프로그램】

◆ 트리를 생성한 후 원하는 node에 대한 path의 합을 구하는 프로그램을 작성하시오.

➢ 프로그램 Flow

- ✓ Root 노드 값을 입력 받는다.
- ✓ Root의 자식 노드를 입력 받는다.
- ✓ 자식 노드 각각의 또 자식 노드를 입력 받는다.
- ✓ (반복 no more node 까지)
- ✓ path의 합을 계산하고 싶은 node 값을 입력 받는다.
  (자신 포함하여 계산)

예



※ Total cost 계산에서 활용 가능