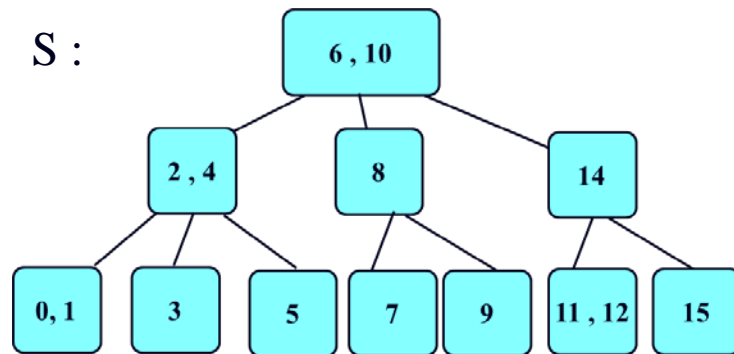


# Programming Assignment #3: Due 12/18 23:59

- ❑ For this programming assignment, you need to implement the **set** class template using a B-tree. Assume that your set class has only the following three public member functions: *count*, *insert*, and *erase*.
- ❑ (10points each) Define and implement the following functions of set class: *count*, *insert*, *loose\_insert*, *fix\_excess*, *erase*, *loose\_erase*, *fix\_shortage*, *remove\_biggest*.
- ❑ (20points) In addition, define and implement a public function named “*show\_contents*” for your **set** class that prints its B-tree contents on the standard output. For example, given the following B-tree of your current set S, the *show\_contents*( ) function of your set class should display the contents of the tree structure as follows:



S :

14	15
	12
	11
10	9
	7
8	5
6	3
	1
	0

# Programming Assignment #3: Due 12/18 23:59

- ❑ (15points) In order to demonstrate that your functions are working properly, implement a main program interacting with the user. Your main program needs to declare a variable of **set** type and then repeatedly perform each of the following instructions until the user enters the “quit” command.

- ❑ insert x – inserts x to the current set(B-tree)
- ❑ erase x – erase x from the current set (B-tree)
- ❑ count x - counts the number of x in the current set (B-tree); returns either 1 or 0
- ❑ quit – quit

Your main program should show the contents of the B-tree before and after each function call.

- ❑ You need to demonstrate that your functions are working properly by doing the following operations :
  - ❑ Perform insert operations at least 10 times. (Try to insert the same numbers a couple of times.)
  - ❑ Meanwhile perform the count operations at least 3 times. (Try to count some inserted number at least twice and uninserted number once.)
  - ❑ Then, perform erase operations at least 10 times. (Try to erase some uninserted number a couple of times..)