

1. Review **True/False**, **Multiple choice** questions
2. Review **Stack** container class along with the time efficiency of its methods (*we had two examples in this topic: check if parentheses are balanced, processing the postfix notation expression*).  
(see HW8 short answer question)
3. Review **Queue** container class along with the time efficiency of its methods (will depend on the implementation: using a *Python's built-in list* or *linked lists* (*you can skip the circular array/list approach*)).  
*We had palindrome check example there, which used both queue and stack.*
4. Review **postfix**, **prefix** and **infix** notations expressions. Be ready to go from one notation to another.  
(see p. 182, Multiple choice question 5 – suggested work in HW8)

Chapter 6 review questions (Lectures 10 & 11)

1. Review **True/False**, **Multiple choice** questions
2. Review recursive definition of *binary search*, *mergesort*, *factorial function*, *Fibonacci numbers*, and others presented in the lectures 11-12 (see the dates 03/11 and 03/16)
3. Review the notion of *good recursive definition* (page 192 in the book)
4. Be sure you can *trace a call of the recursive function* (see short answer questions, page 214 # 4, midterm exam sample #5 )

Chapter 7 review (Lectures 12 & 13)

1. Review **True/False**, **Multiple choice** questions
2. Review definitions of *tree*, *complete tree*, *full tree*, and *BST*.
3. Review example of a tree: *expression tree*
4. Review the code of the **BST** class, how insertion, deletion methods are working, along with the time efficiency of its methods.

Chapter 8 review (Lectures 15, 16 & 17)

1. This is an introduction to C++ chapter. Review it to be able to write code in C++.
2. Review **True/False**, **Multiple choice** questions

Chapter 9 review (Lectures 18 & 19)

1. This chapter is devoted to C++ classes. Know the basic syntax and semantics of class definition.
2. Review the code of the **Rational** class.
3. Review **True/False**, **Multiple choice** questions