|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***MODULE NUMBER*** | | 03 | | | |
| ***TOPIC(S)*** | | Recursion | | | |
|  | | | | | |
| **--- PRECLASS ---** | | | | | |
| **SHORT VIDEO** | Recursion: <https://goo.gl/sOQUEZ> | | | |  |
| **BOOK** | GTG Chapter 4 (until p. 167, skip section 4.1.4) | | | |  |
| **QUIZ** | ADSA Quiz Module 03 at <https://goo.gl/forms/lmYbcsCDUUnw0drA2> | | | |  |
| **TO-DO** | PyCharm: Module03 >> part00 >> power\_recursive   * Complete the implementation of **recursive\_power()** * Complete the implementation of **non\_recursive\_power()** | | | |  |
| **--- PROBLEM SET 1 (Tuesday September 13th) ---**  PyCharm: Module03 >> part01 >> binary\_search\_recursive  Design and implement a function that checks whether an array (= Python list) is sorted in ascending order [ **is\_sorted()** ]  Design and implement a recursive binary search algorithm that looks for an element in an array [ **binary\_search()** ] | | | | | |
| **--- POST-CLASS 1 ---** | | | | | |
| **TO-DO** | Complete Problem Set 1 in PyCharm | | | |  |
| **--- PROBLEM SET 2 (Tuesday September 20th) ---**  Improve your StudentManager class of Module02 under Module03 >> part02 by adding the following methods:   * Change an existing grade [ **update\_grade()** ] * Calculate a “strange” GPA obtained by multiplying all grades (with recursion if possible) [ **get\_strange\_gpa()**] * Calculate the number of courses for which the student achieved B+ or more [ **get\_count\_of\_above\_bplus()** ]   Cut & paste the StudentManager class implementation that you completed in module 2, then complete the implementation of the 3 new methods. | | | | | |
| **POST-CLASS 2** | | | | | |
| Complete Problem Set 2 in PyCharm | | |  |  |  |