# 1 Laboratory P3: Software Development Planning/Coordination

## 1.1 Recommended P3 Session Group Discussions

Each group should gather the modules from sessions P1 and P2 and plan the remaining software development tasks:

1. Review the suggested code architecture in the specification document. Consider the major sections of code necessary to accomplish the tracking task and decide if the suggested architecture makes sense. Feel free to use a modified architecture from that suggested in the specification document.
2. Develop the high level algorithm in Python for each section of code. Determine the inputs you will need and the outputs each section will generate. Feel free to break up your section of code into smaller steps which will logically be developed into separate functions later on.
3. For each major section of code:
    a. Describe the function(s) for each block,
    b. Develop the main (calling) function based on the functions from each section,
    c. IMPORTANT: consider how you will structure your software to make development and debugging easy. Use STK to provide test case answers. The use of debugging tools and how all inputs are to be integrated should also be agreed upon.

## 1.2 P3 Report requirements

Each group will submit a short development plan and the outline source code for the main program with major section function calls. Your development plan (Word file) should have the following sections:

1. Outline the algorithm used in the main script containing high level function descriptions from each section. Describe the interface between the main program and major sections (i.e. what data does each section need and what outputs will it produce? - similar format to P1 and P2 function descriptions).

2. A brief test plan, which answers: How will you know if your output is correct? The test plan should outline test cases to be used and the use of log files for partial results, error status checking in the code, etc.

The main script source listing to be provided should:

- Correspond to the algorithm outline given above
- Identify the major code sections as subroutine calls in a format similar to the subroutine descriptions for P1 and P2 sessions, except that the subroutines are arranged in a logical sequence for the tracking task.

- A suggested first step is that the main function should create all 32 instances of Satellite class objects representing or containng the TLE data on the GPS spacecraft. Then for an arbitrary 30-minute time window (say within one or two days of the TLE refepochs), determine which of the satellites are in view of ARO (which is contained in a Station instance). Finally output a list of in-view spacecraft similar to the AOS/LOS listing given in the development specification. The listing should then be compared with a known correct list (from STK).
- Note the main script is not in a completed state at this point, but all team members should have a copy to use or refer to for development of their code.