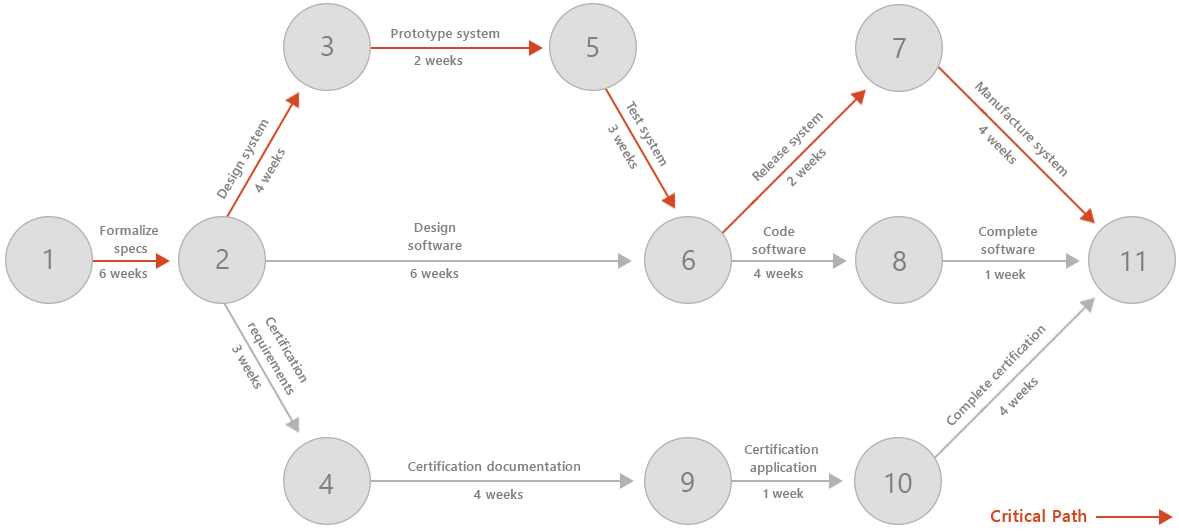
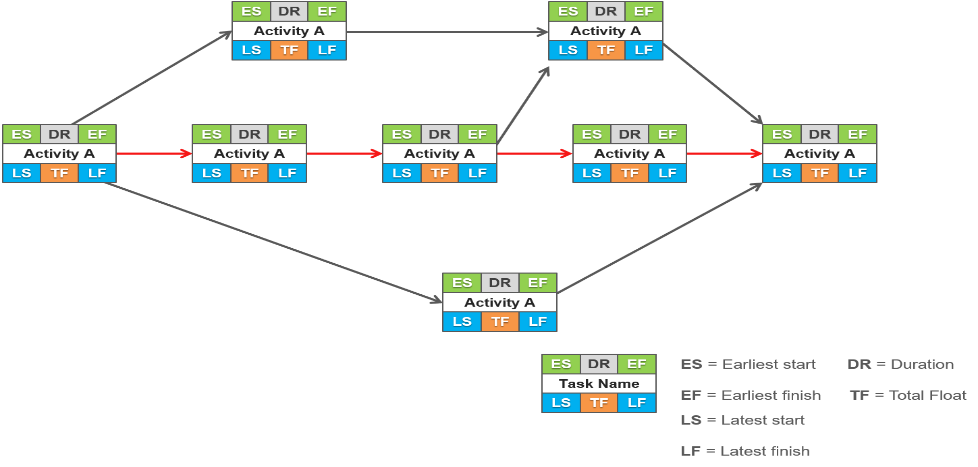
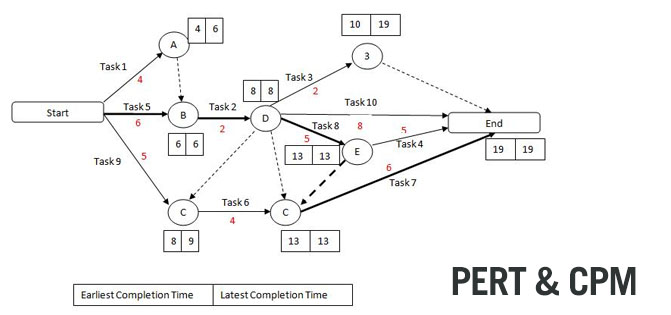
Homework: Project Management Pert







**This is the default project to be insert in code**

General

* Build a Pert or CPM system (A project management system) on a graph class in Python OOP, demonstrating the essential facts and functionalities of managing project.
* The graph must begin with one start node and end with one finish node.
* You can add any methods helper to the project.
* The homework is for two students. Deliver it twice to the Moodle site (Two student per work with the name’s student in the work).
* The solution must be on Python file or files(.py).
* Give meaning names to the variable.
* Write remarks before any method, and explain what going on each method.
* Logger is a must for each method. All logs will be written to a file. And eventually read the file for Python context (for convenience).
* Unit Test is not mandatory (no points) but nice to have.

================================================

1. Initializes a graph object. If no dictionary or None is given, an empty dictionary will be used. For testing use the default graph that draw above.
2. Build method that add activity to the project, Activity must have at least two fields: name, duration.
3. Build method to find isolate activities. An activity without following or ascending another activity.
4. Build method that return slack's time for each activity in descending order. Don’t show the critical activities in the list (By definition, critical path has activities with a zero-slack time)
5. Build method that return the sum of the slacks time in the project
6. Overwrite \_\_str\_\_ method for any project’s class, with clear detail.   
   The \_\_ str\_\_ also include a property’s class that show the duration of the project.
7. Define iterator that iterate over all the nodes inside the “Pert” class.
8. Build method that find “Critical path” of the project (Showing the edges of the critical pass with their length (20 point).
9. Build method that build and return dictionary.  
   The dictionary contain each task in the “Critical path” as a key, with a value that represent maximum shorting time duration of the task that can be achieve without leave the “Critical path”.   
   Remark : Every task must be at least 1 time duration (you can't sorter the task to 0 duration).