

CS 302 Introduction to Data Structures

University of Nevada, Las Vegas

Spring 2018

Assignment 6

Due: Saturday, March 10, 2018

Background. Once you have graduated you may write code in a team of software developers. Further into your career, you may be put in charge of seeing a project through as the project lead. In this case it is important to keep track of deadlines.

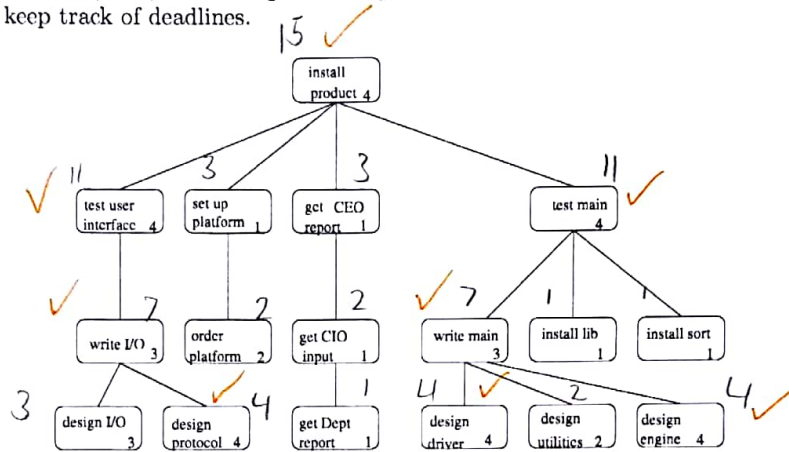


Figure 1: A typical software project

Scenario. In Figure 1 you can see a typical software project. The numbers give the time required for the task in weeks. In this case the dependencies among the tasks are tree-like, this is not always the case in practice but for this assignment we will assume that we have such a structure. Some tasks are critical, which means that if those tasks are delayed then the entire project will be delayed. In fact, all tasks on the path "design protocol", "write I/O", "test user interface", "install product", are critical, whereas task "get CEO input" is not.

Problem. You should write a program that prints out a schedule of tasks in a feasible order and which gives the the earliest completion time for each task and says whether a task is critical or not. Your printout should look like this:

PROJECT SCHEDULE

design I/O	not critical	3
design protocol	critical	4
write I/O	critical	7
test user interface	critical	11
order platform	not critical	2
...		

Implementation. The tree should be implemented as a left child, right sibling pointer structure. There is a .h file posted on the announcement page.

Report. Submit the following files electronically:

1. The actual program.
2. Output for the tree in Figure 1. (Text file).
3. A memo about the project's deadlines to your colleagues, with critical tasks emphasized. (PDF file).

Email these as attachments in an email to deshmk1@unlv.nevada.edu. Subject of your email must be "Assignment 6", <your name>, <your student ID number>.