Introduction to Software Engineering
Final Exam Part2

Spring 2020

Please submit your solution via ODTUClass before the deadline. Submit a SINGLE word or pdf file. Your word or pdf file may have embedded images such as a) photos of your drawings of Z schemas and/or charts; b) screenshots or output images of a program like dia/ppt/visio etc.

This is a takehome EXAM.

1. You are required to follow **academic integrity**.
2. NO COLLABORATION is allowed.
3. Note that any exam is an act of **trust** between the students and their **classmates**, as well as between students and instructors.

1 Question1 (20)

We are specifying a dental appointment clinic software. The appointments are taken as first come first serve policy. Not all the dentists see patients, some of them are on vacation.

Assume the sets Patient and Dentist are given. i.e. [Patient,Dentist]

The state of the system is specified as follows:

Clinic

$$appointment : Patient \pfun Dentist$$
$$list : \seq Patient$$
$$onduty : \power Dentist$$

$$\ran(list) \subseteq \dom(appointment)$$
$$\ran(appointment) \subseteq onduty$$

In this schema I used a sequence to represent the patient queue. There is no capacity limit on the queue. The set of dentists who are not on vacation are represented with *onduty*.

Specify the operations *MakeAppointment* and *NextAppointment*.

The operation *MakeAppointment* will take a Patient and a Dentist as input. It puts the patient into the queue. The patient is assigned to the dentist given as input. If the patient has already been assigned to a Dentist previously, that previous assignment is removed. i.e. the new dentist assignment overrides any existing dentist assignment.
Error case: If the dentist given as input is not onduty, give an error and do not take the appointment (do not change the state).

The operation *NextAppointment* will remove the patient from the waiting queue (i.e. *list*) and return the patient as well as the doctor assigned to that patient. The value of the *appointment* does not change.
Error case: Give an error message if the list is empty.

2 Question2 (12 points)

Consider the following table. This table sets out a number of tasks, their durations and dependencies.

Task	Duration(days)	Dependencies
T1	10	-
T2	15	T1
T3	10	T1,T2
T4	20	-
T5	10	-
T6	15	T3, T4
T7	20	T2
T8	35	T7
T9	15	T6
T10	5	T5,T9

a) Draw pert chart where activities are on arcs. (8 points). You will loose points if activities are not on arcs.

b) What is the shortest possible completion time (1 points).

c) Give a Gantt chart for this project. Assume that you have the shortest possible completion time that you have calculated above.(3 points)