VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

MATHEMATICAL MODELING (CO2011)

Assignment

"PETRI NETWORKS"

(version 0.2)

Advisors: Nguyễn Minh Văn Mẫn
Nguyễn An Khương
Nguyễn Tiến Thịnh
Students: Trần Phạm Minh Đăng - 2052070.
Nguyễn Việt Thắng - 2052719.
Nguyễn Lê Thanh Phúc - 2052656.
Đào Châu Gia Bảo - 2052867.
Nguyễn Phước Nguyên Phúc - 2053342.

HO CHI MINH CITY, DECEMBER 2021

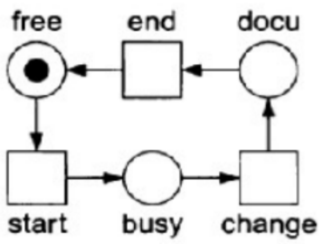
Contents

1 Member list & Workload

No.	Fullname	Student ID	Problems	Percentage of work
1	Trần Phạm Minh Đăng	2052070	- Question 1a, 3, 4 & 7	20%
2	Nguyễn Việt Thắng	2052719	- Question 1b, 6 & 7	20%
3	Nguyễn Lê Thanh Phúc	2052656	- Question 2, 6 & 7	20%
4	Đào Châu Gia Bảo	2052867	- Question 2, 7 & Check	20%
4	Nguyễn Phước Nguyên Phúc	2053342	- Question 1a, 3, 5 & 7	20%

2 Petri Nets - Assignment on Modeling

2.1 Given the Petri net N_s modeling the state of the specialist, as in Fig. 5.21



A Petri net modeling the state of the specialist.

Figure 5.21: *The Petri net of the specialist's state*

a) Write down states and transitions of the Petri net N_s [**1 point**]:

- There are 3 states: (1) the specialist is free and waits for the next patient (state "free"), (2) the specialist is busy treating a patient (state "busy") and (3) the specialist is documenting the result of the treatment (state "docu").
- There are 3 transitions: First, the specialist starts with the treatment of a patient (event "start"). Second, the specialist finishes the treatment of a patient and starts documenting the results of the treatment (event "change"). Third, the specialist stops documenting (event "end").

b) Could you represent it as a transition system assuming that

(i) Each place **cannot** contain more than one token in any marking; and

There is maximum one token in total in all markings:
We denote a triple (x, y, z) with x specifying the number of tokens in place "free", y in place "busy", and z in place "docu". Therefore m(free) = x , m(busy) = y , m(docu) = z and x + y + z = {1,0} .

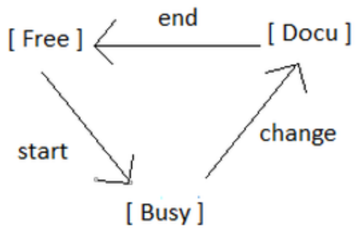


Figure 2

(ii) Each place may contain any natural number of tokens in any marking. [**1 point**]
Use the same idea. We have: x, y, z ∈ N.

2.2 Figure 5.21 of net N_s is made by information of Specialist and Event data above.

Define N_{P_a} as the Petri net modeling the state of patients. By the similar ideas,
a) explain the possible meaning of a token in state "inside" of the net N_{P_a} [**1 point**]
b) construct the Petri net N_{P_a}, assuming that there are five patients in state "wait", no patient in state "inside", and one patient is in state "done". [**1 point**]

a)
A token in state "inside" represents there is a patient treated by specialist. It shows that a token in state "wait" has just moved, or the transition "start" has just been firing. On another hand, the transition "change" is now enable.

b)
Five patients in state "wait" are represented by 5 tokens. And a token in state "done" represents a patient. There is no patient in state "inside" so no token there.

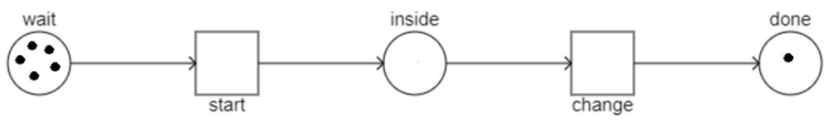


Figure 3

2.3 Determine the superimposed (merged) Petri net model $N = N_s \oplus N_{Pa}$ allowing a specialist treating patients, assuming there are four patients are waiting to see the specialist / doctor, one patient is in state "done", and the doctor is in state "free". (The model then describes the whole course of business around the specialist). [1 point]

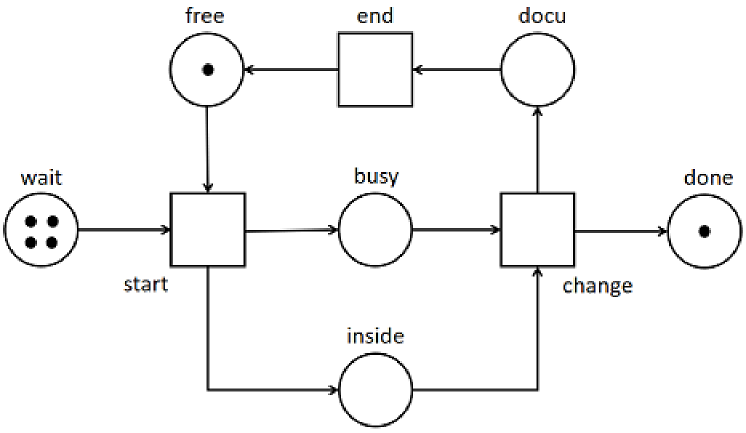


Figure 4

We can see that state "free" from N_S and state "wait" from N_{Pa} both have $(N_S, [free])[start >$ and $(N_{Pa}, [wait])[start>$ so we can merge this two states. Do the same with state "busy" and state "inside". Finally we have the superimposed Petri net model N above.

First of all, the specialist stay in state "free"(Figure 4). If there is at least one patients in state wait, one specialist will "start" and come to state "busy" while one patient go to state "inside" (Figure 5).

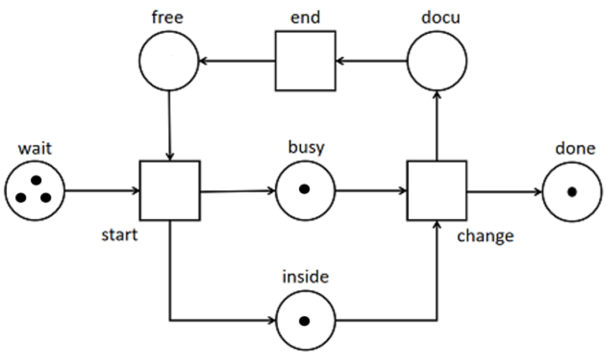


Figure 5

Then after the treatment, the specialist and patient will come to "docu" and state "done" respectively (Figure 6).

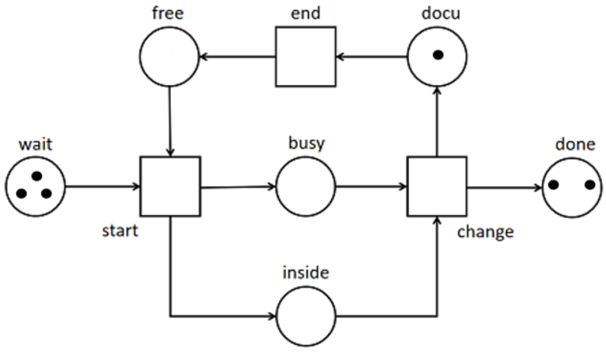


Figure 6

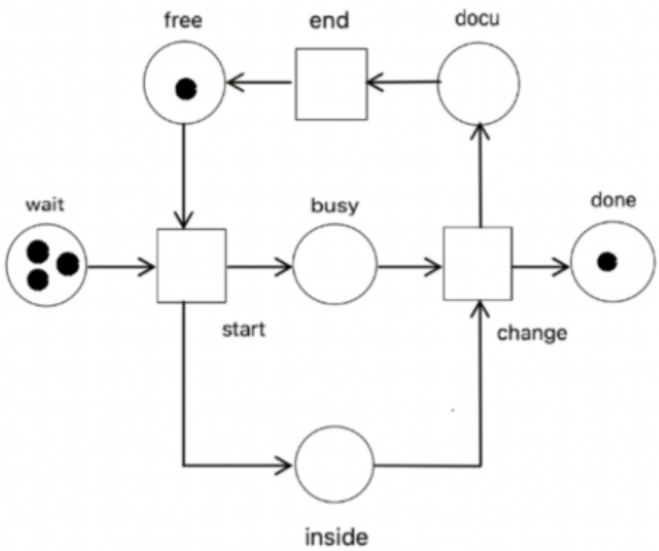
After the "docu" state, the specialist will back to state "free" and keep working if there exist any patient in state "wait".

2.4 Consider an initial marking $M_0 = [3.wait, done, free]$ in the grand net $N = N_s \oplus N_{P_a}$. Which markings are reachable from M_0 by firing one transition once? Why? [1 point]

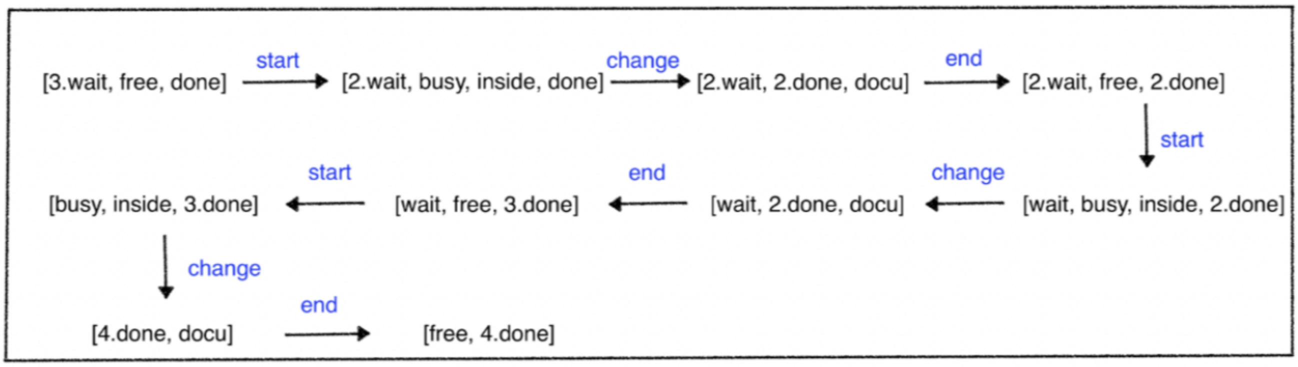
To begin with, a marking M is accessible from the initial marking $M_0 \iff$ there is a series of enabled transitions that fires from M_0 to M. $[N, M_0>$ denotes the set of attainable marks of (N, M_0). As a result, in Fig.4, the accessible marking of M_0 following the sequence is [wait, done, docu].

Following that, assume the initial marking as M_0 = [3 wait, free, done]. Following the definition of reachable marking, we can now fire a single transition to acquire the set of all reachable markings from an initial marking.

Finally, we express the collection of accessible marks as a graph, which we call the **net's reachability graph**. Its edges correspond to the transitions that move the net from one marking to another, and its nodes correspond to the accessible markings. The reachability graph is depicted in the image below:



a, the Petri net system of specialist's and patient's states,
M = [3.wait, free, done]



b, Reachability graph

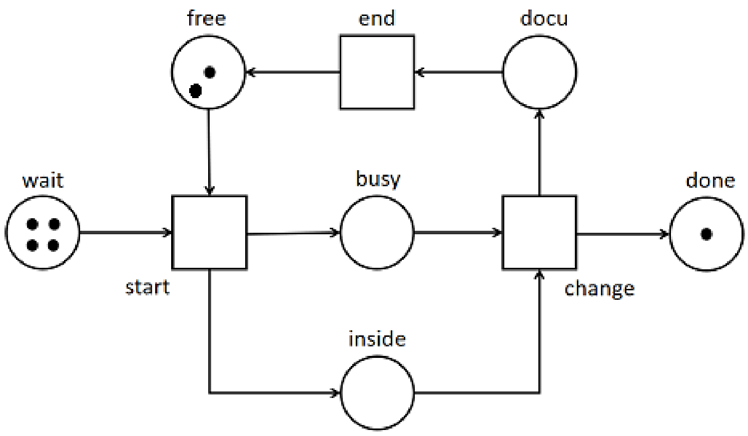
Figure 7: The Petri net system N and the reachability graph

Observing **Fig.7,** we may deduce from the marking M = [3 wait, free, done] that there are a total of **10** attainable markers.

2.5 Is the superimposed Petri net N deadlock free? Explain properly. [1 point]

The superimposed Petri net N is **not deadlock free** because at some reachable markings, no transition in enable, which mean the Petri net N stops running (dead). For example, When all the patients already treated and all the specialist already finished their documents, so all the patients are in state 'done' and all the specialists are in state 'free', so there isn't any token left in state 'wait' and state 'inside', therefore the transition 'start' and 'change' can't fire, the Petri Net will dead.

2.6 Propose a similar Petri net with two specialists already for treating patients, with explicitly explained construction. [1 point]

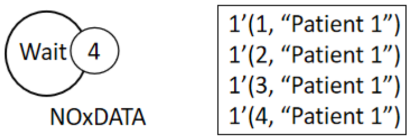


There was no trouble if the model from the start with only **one** patient and **one** special list, but there will be some complications like the example above if there are **more than one** specialist and patient so there are certain challenges that we had discovered:

- First, we are unable to **determine between specialist tokens and patients tokens** *(all of them is drawn by one black dot)*. In practice, however, each expert has a unique skill set and is in charge of certain health issues. As a consequence, the selection of a professional to aid a patient is not arbitrary.
- Second, the problem could be that because patients are **submitting orders incorrectly**. Patients must enter the room in the correct sequence.

As a result, we require a new tool to tackle these issues. It also is called as a **Colored Petri net** *(CPN)*. CPN is a mathematical model of Petri net. Instead of using black dot to symbolize a token, **each token in CPN has connected data (token color)**. Although token color is an **abstraction notion**, locations in **CPN only have one style of token color**. To keep it short, we developed the model with two specialists and four patients. We have two specialists, thus when marking M0, in places free there will be two color tokens.

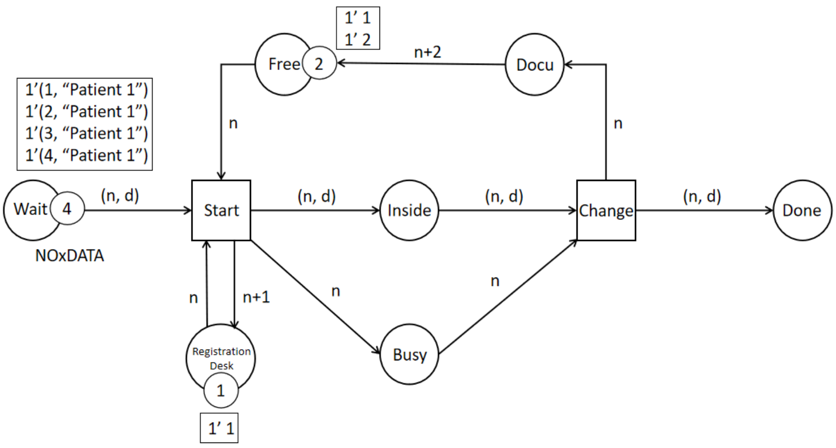
This location demonstrates that each token has a data value of **NO** associated to it, and there are presently two tokens in place FREE with data values of 1 and 2. At M_0, the same was true for four patients.



All of 4 waiting patients have type **NOxDATA**, where **NO** is the order of them, 1 for the first patient and 2, 3, 4 for the next. **DATA** is name of their symptoms or problems (information of patients).

We will now look at a little **more complicated CPN**. It is based on the Petri net model, which has already been mentioned in previous sections, However, it is rapidly taking over the **order of patients entering**. There is a new location named the **Registration desk** and it has a token with the value of 1 at the starting point. Only transition Start can be enabled at marking M_0. When the transition Start fires, it takes tokens from the registration desk and places binding n = 1, token (1, "Patient 1") in place Wait and token 1 to be placed Free.

When the transition **Start** happens with an enabled binding, tokens are withdrawn from the **Registration desk** based on the arc expression result, but they are instantly replaced by new tokens with the same token colors, and the value of this token is increased by 1. It implies that once the receptionist calls number 1 into the room, the data of token color is raised by 1, and the expert calls the second patient in the following round.



Steps	Binding elements
1	(**Start**, $<$ n = 1, d = "Patient 1" $>$)
2	(**Inside**, $<$ n = 1, d = "Patient 1" $>$), (**Change** $<$ n = 1 $>$)
3	(**End**, $<$ n = 1, d = "Patient 1" $>$)

The token will be returned to place FREE with value n+2 when transition END is fired, which is another difference between this model and the old one. The number 2 is the number of specialists; after assisting patients, the order of specialists is increased by two. It ensures that two specialists enter the room in a timely manner to assist patients.

2.7 Write a computational package to realize (implement) Items 1,2,3 and 4 above. You could employ any programming language (mathematical-statistical like R, Matlab, Maple, Singular, or multi-purposes language like C or Python), that you are familiar with, to write computational programs. The programs of your package should allow input with max 10 patients in place "wait" [2 point]