DEPARTMENT OF COMPUTER SCIENCE AND TECHNOLOGY COURSEWORK ASSESSMENT DESCRIPTION 2021/22

Module Number: 701026

Title of assessment: Dependable Systems

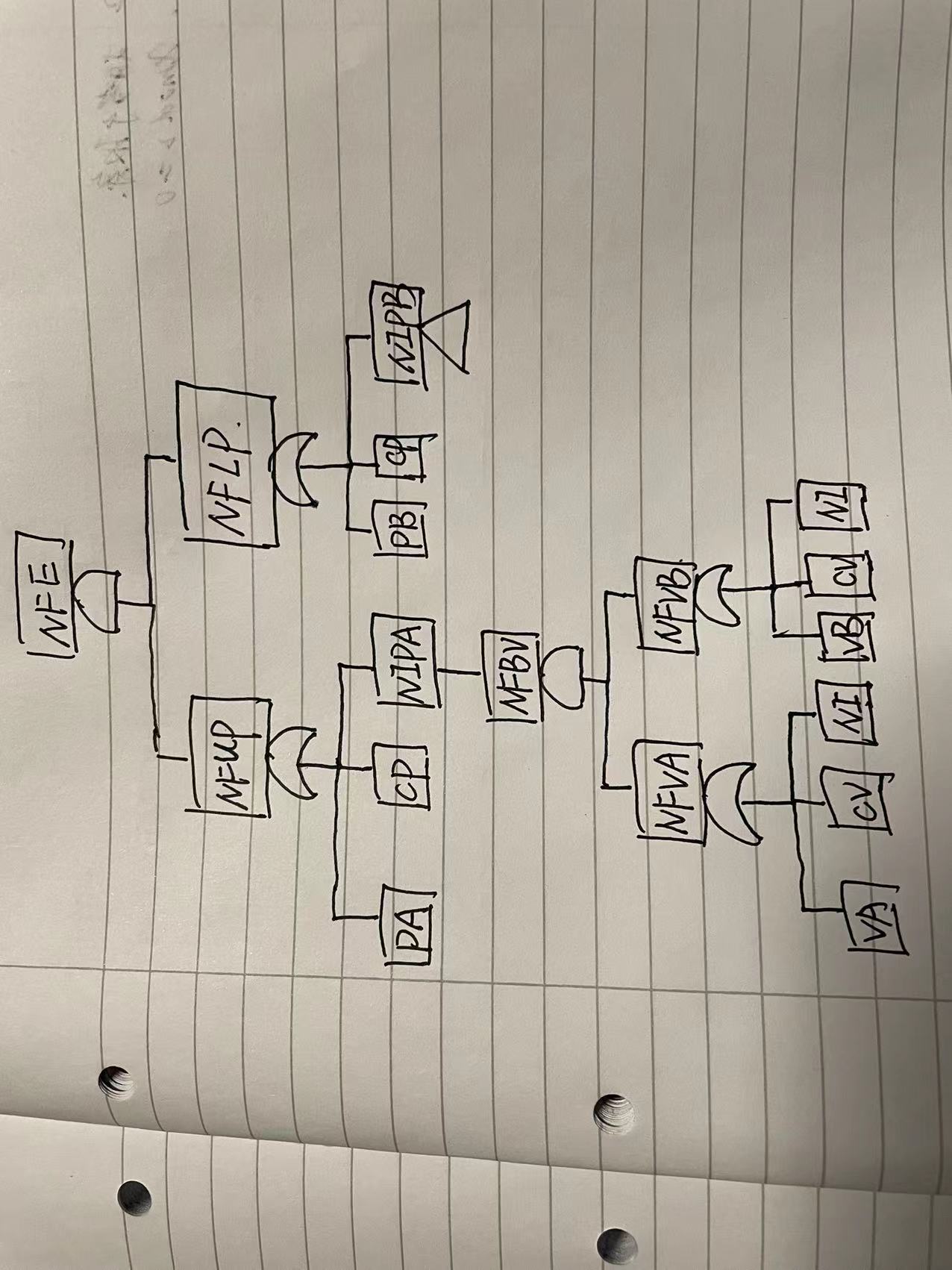
Module Title: Real-Time Dependable Systems Lecturer: Professor Yiannis Papadopoulos

Student Name: CHUNCHUAN LI

Student Number:690085/202119900

**Q1**

(i)



(ii) Calculate the minimal combinations of basic faults (i.e. minimal cut sets of fault tree) that cause NFE.

NIPA and NIPB have the same cutsets

Because x+x=x, x.x=x, x(y+1)=x

NFE=(NFUP).(NFLP)

=(PA+CP+NIPA).(PB+CP+NIPB)

=PA.PB+PA.CP+PA.NIPB+CP.PB+CP.CP+CP.NIPB+NIPA.PB+NIPA.CP+NIPA.NIPB

= PA.PB+PA.CP+PA.NIPB+CP.PB+CP+CP.NIPB+NIPA.PB+NIPA.CP+NIPA.NIPB

= PA.PB +PA.NIPB+CP+NIPA.PB+NIPA.NIPB

=PA.PB+CP+PA.NFBV+PB.NFBV+NFBV.NFBV

= PA.PB+CP+PA.NFBV+PB.NFBV+NFBV

=PA.PB+CP+NFBV

=PA.PB+CP+(NFVA.NFVB)

=PA.PB+CP+[(VA+CV+NI).(VB+CV+NI)]

=PA.PB+VA.VB+VA.CV+VA.NI+CV.VB+CV.CV+CV.NI+NI.VB+NI.CV+NI.NI

= PA.PB+VA.VB+VA.CV+VA.NI+CV.VB+CV+ CV.NI+NI.VB+NI.CV+NI

=PA.PB+VA.VB+CV+NI

As we can see that PA.PB, VA.VB, CV, NI are minimal cutsets

(iii) Identify single points of failure in this system.

PA,PB,CP are single point failures, because PA and PB and CP are directly causing the top event and they are also cutsets and they are single nodes.

(iv) Assume that there are no failure detection and control measures in this system. Give examples of two measures that could be incorporated in the monitoring and control software of the Controller to improve the design of this system.

The first measure is that we can use controller to manually control one pump or valve and let another one automatically run, the advantage is that it would decrease the rate of hazardous to the Engine

The second measure is that we add one more controller in case that the only controller which controls both valves and pumps stops.

**Q2**

1. If k1 and y1 are the values of k and y just before the execution of the while statement, what conditions must k1 and y1 satisfy to cause the unsafe condition “y<0” being true and false must be examined. Explain how you have derived these conditions.

If k>3 is true, it might cause iterations

Yn=y1+4n<0 =>Yn=y1+4n =>y1+4(k1-3)<0 =>y1+4k1<12 and k1>3

kn=k1-n => n=k1-kn

kn=3

if k>3 is false, so k1<=3

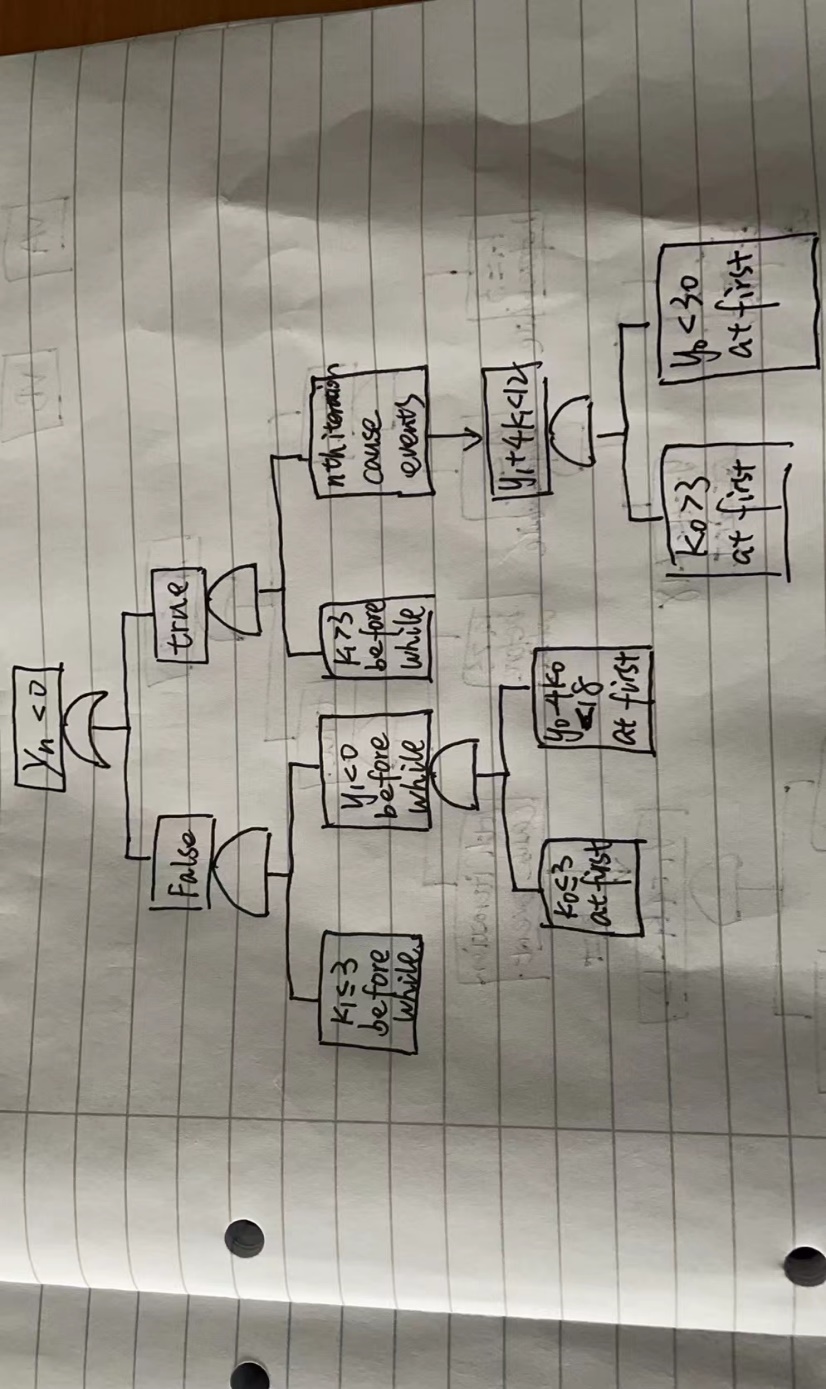
because y1 is just before the execution of the while statement and y<0,so y1<0

the result is k1<=3 and y1<0

1. If k0 and y0 are the initial values of k and y just before the execution of the first assignment (i.e. y=y-4\*k-18), what conditions must k0 and y0 satisfy to cause the unsafe condition “y<0”) being true and false must be examined. Explain how you have derived these conditions.

If k>3 is true, y1+4k1<12=> y0-4k0-18+4k0<12, the result is y0<30 and k0>3

If k>3 is false,k0<=3,y0-4k0-18<0=>y0-4k0<18,so the result is y0-4k0<18 and k0<=3

(iii)

(iv)

We can add some conditions before execute the while sentence and to be pretty sure that y>=30 before into the while loop

Int k,y;

If(y<=30 or k>3){

Break;

}else{

Y=y-4k-18;

}

While(k>3){

k--;

y=y+4;

}