

ζ»»(Client)°·ÎÆSerrer

2,Ó»ÓÒì¼Ä»£±ÊÊÃµ·ÊÓÊÃµÃ×£CLIENTφ° CLIENTφ;Ô2ì¼Ê»¹ÌÒì»¶Ì´Ó»£SERVERφ£SERVERφ;µµ;Ã×Ô2ì¼Ê»µ´Ó×  
Ò±Ð°ÊÐÔ´½Ð;ÎÄÐµÒÌ´Ó°²ì¼ÄÊ»Ô¼µÎÐ£ÎÄÊÓÃÔ,ÎÊµ¶Î£Ð°µ»¹(SEMAPHORE)

Ð°µ(SEMAPHORE)

ÃÌ´Ó»ÓÔ,ÓÔÓ¹µÐ°µ;Ð°µÓ2,×Ì£S1°S2;µ´Ó»¿Ò½Ê´Ó×ÒÊ£ÆÐ°µ´Ó×ÌS1£µ´Ó»ÔÃÓÔÐÃ´Ó×ÒÊ£ÆÐ°µ´Ó×ÌS2;  
Ð°µµ×Ì×»¿ÒÊ£;S1;S2;°;S2;S1;  
µÓ»,¼ÄÊ»·Ô,´Ó×ÒÊ£¼ÄÊ»½ÎÐ°µ·³Ò,Î£; ARE YOU OPEN;  
ÊÐ°µ´Ó×ÌS1£Ô,Ð°µ½×»µ×ÌS2£²ÇÌ,Ê·³ÊÊÐ°µ¼ÄÊ»»´Ò,Î£;OPEN;µ;  
ÊÐ°µ´Ó×ÌS2£Ô,Ð°µ»µÎ½Ê;CLOSED;µ; µ´Ó»Î³´Ó×ÒÔ°£,´Ó»½ÎÆÐ°µ·³Ò,Î£;READY;(ÒÇ¾Ä£¿Ò½Ê´Ó×Ò);Ð°µÊµÎ;  
READY;Ô°£¼×××ÌS1;

¶ÎÄÐ©©Ð°µ(SEMAPHORE)©©µÃÊ

¹¼1(DOCUMENTATION 1)ÔÃ½ÔµÐ°µÎÎÃÊ;ÆÔ°££  
;¶ÎÄÐ°µµÊÓ¿Ãµ±Ê·(IDENTIFIER)°×Ì(state);  
;Ô¼¼Ä±(Priority List)  
;ÎÐÎ(Communication Diagram)  
;×Ì×»Î(State Transition Diagram)  
; `Receive Procedures' ©©ÊÓÃÊµÇÐ°µÊµÎ£; Ready;µ;Are\_you\_open;Ô°£Ð°µÊÓ»µÎ; Ô¼¼Ä±ÔÊÐÔÊÔÎÐ°µÓ  
¿ÃÔÔÊ¼ÄÊµ¶Î£¶ÎÐ°µÔÃÊÐµ´ÃÔÐÎ;Î¼1ÔµÔ¼¼Ä±ÔÊ£·ÆserverµÎÎµÔ¼¼,Ó¿»»clientÎµÔ¼¼;ÔÊ£server(2)µÎÎµÔ¼¼,Ó  
server(3)µÎÎµÔ¼¼;

¶ÎÄÐ©©¿»»(CLIENT)©©µÃÊ

¹¼2(Document 2)Ã½Ôµ¿»»µÎÎÃÊ;CLIENTÄÐµÒ,¶Î¿Ò´ÓÔÊÔ×ÌÔµÒ,SA£SB»SC;  
;µÃ,CLIENTÃÔ·³´Ó×Ò£²ÇÔÔ¹µSEVER(¼¼´Ó)Ô²ÔÃÔ·Ã,CLIENT(¼¼¼ÄÊ»µ´Ó×Ò£Ô,CLIENT´Ó×ÌSA;  
;µÃ,CLIENTÎÎ·Î,SEVER£Ô,CLIENT´Ó×ÌSB;  
×Ô£CLIENT±Ð¾¼¹Ð°µSEMAPHORE²Ã·SEVER;  
;µÃSEVERÔÔÔÐÃCLIENTµ´Ó×Ò£Ô,CLIENT´Ó×ÌSC;  
Ô,¿»»CLIENTµ×Ì×»¿ÒÊ£;SA;SB;£;SB;SC;£;SC;SA;µ¿»»´Ó×ÌSB£Ê¿Ò½ÊÐ°µ·ÃµÎ£;CLOSED;µ;ÔÊµ;CLOSED;Ô°£Ê±  
Ðµ´Ò¶ÎÊ¼;(i WAITING PERIOD);²ÃÔÔ·³Î£;are\_you\_open;£,Ð°µ;Ê¼¿»»ÊµµÎ½Ê;CLOSE;¶Î;OPEN;£ÃÊ¾×±µ×ÌSC£Ô´Ð  
°µÓ¹µSEVER(¼¼´Ó)·³´Ó×Ò¼¼Î;S£Job;Î;ÔÊ£,SEVERÔÐ´Ó×Ò;µ´ÓÔÐ¹±Ô°£,SEVER½ÎÊ·²,Î£  
;Ô,Ê£;READY;(¾Ð)£,Ð°µ;  
;ÃÔ,Ê;C£READY;£,CLIENT;CLIENTÊµ´Î´Ó×ÌSC×±µ×ÌSA;  
Ã¿Ò¼¶Î´Ó»ÊÃÎ»(Î¹Ô)µ£Ê¶ÔÊÊµµÊ°´Ó×Ò¶Î¿Ò;Ô³;  
¾ÄÃÊ£ÊÊÔ²»³;Î;ÊÔ;µÇÐ;

ÎÐ

¹¼3(DOCUMENT 3)ÔÃ½ÔµÎ¼½¹;ÆÔ,³ÊÓÎÎÄÐ¼Î,³Ô°½ÊÔÔ¼¼µ¹;ÔÎÎÃ±(MESSAGE LIST)Ô£Ã¿ÒÔµÃÔÎÎÄÐµÊÃÊÃ  
ÔÎÎÔÔ,±Ê·£Ô,Î·³Ô£Ô,Î½ÊÔ£ÔÊ»°»ÎÎÃÊ;  
Î·³ÔÔÊ¿T·³±Ê·ÎµÎ£Ô½ÊÔÔÊ¿T£1´Ã,Î£¼ÔÐ  
;RECEIVE PROCEDURE;A  
ÊÔÔÔÊ¿T£Ô¶Î,Î·³ÔÔ½ÊÔ·³Î£Ô½ÊÔÔÊ¿T£1´ÃÊÔÔÐÎÎ£ÆÎ°ÊÐ×Ô·³ÔÔ½ÊÔÔÔÎÎÃ±ÔµÊÐ;

×ÊÎA

Ô,¾²ÇÓÎ(LAN)ÔÊ¿Ô°°ÎÃ¶Î£

Object:CLIENTφ£Client.State=SA,

Waiting\_Period=2,Number\_of\_Servers=1

Object:CLIENT $\phi$ ,Client.State=SA,  
Waiting\_Period=1,Number\_of\_Servers=1

Object:SERVER $\phi$

Object:SEMAPHORE $\phi$ ,Semaphore.State=S1

3/4ÓÍ£Ö.3íÁíí£

At time 1:CLIENT $\phi$ sends a message with the  
identifier "Are\_you\_open?"

At time 2:CLIENT $\phi$ sends a message with the  
identifier "Are\_you\_open?"

At time 4:SERVER $\phi$ sends a message with the  
identifier "Are\_you\_open?"

At time 5:CLIENT $\phi$ sends a message with the  
identifier "Are\_you\_open ?"

î¼£Ö££¶ÓÐ°µSEMPHORE $\phi$ £·Î£°¿»CLIENT $\phi$ °CLITENT $\phi$ ÔÊ¿±Ö(ÖµÊ¿6)ÖÐ¶íÊµµíí£·3µíí,´ÓÊÃ×î¼(»Ö)×±µÊÃ×î;

îA.1

ÔÊ¿££ È CLIENT  $\phi$ Êµíí C\_JOBî¼½»·ÊÊÃÊ£½´°ÐÔî¼5(Documentation 5)

îA.2

ÔÊ¿££ È CLIENT  $\phi$ Êµíí C\_JOBî¼½»·ÊÊÃÊ£½´°ÐÔî¼5(Documentation 5)

îA.3

Ê·ÊíÁÊ¼£

îÔÊ¿8£SERVER $\phi$ ·ÉÓ±Ê·jReadyîµíí

îÔÊ¿10£CLIENT $\phi$ ÊµÓ±Ê·jC\_JOBîµíí

îÔÊ¿12£SERVER $\phi$ ·ÉÓ±Ê·jReadyîµíí

Çî°î¼4ÖµÊ¿±(ÖµÊ¿13)

×ÊîB

3/42ÇÓ±ÀÖ£Ê°°Á¶Ð°µ©©·Î££(SEMAPHORE $\phi$ £SEMAPHORE $\phi$ , SERVER $\phi$ £SEVER $\phi$ )

¶ÖÃ¿»»CLIENT,Number\_of\_Server=2îíÁÃÊÓ2,´Ó»£î¼2Ö¶íÀÐµ¶ÖÐÖ×Ð,±î¿Ôî¼6ÖÖµ,±:

`Receive Procedures`:C\_Job and Wait

Ôî¼6ÖÃ»¿ÖÖµÔÊ¿0µÇ¿µÃÊî

ÔíÁÊ¿£½ÊµííÁÐí£

At time 0:CLIENT $\phi$  receives a message "C\_Job"of a user.

At time 0:CLIENT $\phi$  receives a message "C\_Job"of a user.

At time 4:SEMAPHORE $\phi$  receives a message "Ready".

3/4CLIENT¶íÀÐµ,±£ÔÃÖµ3/4ÓÍLANÖ½»·ÊÊÃ£

Ôî¼6Ö±³ÖÊ´°î

×ÊîC

×ÊîBÖ¶Ö¶íÀÐCLIENTµ¶Ö,±£¶ÓÓ¶,Ð°µ©©·Î£¶µÀÖµ3/4ÓÍÀ£ÖÖ3/4Ê21Áî

×ÊîC.1

±¶íÀÐCLIENTµ¶Ö(¿¿î¼2)£Êµ3/4ÓÍLAN(3/4Ö¶,Ð°µ£·Î£¶)µ¹ÃÊí£





(15.3-2)

Receive Procedures 1/2 E13

```

procedure C_job (User,Client,Document)
begin
  if State =SA
  then State ;SB
    Send ("Are_you_open(Client,SEMAPHORE (Semaphore Index))")
  else
    if State =SB
    then Send ("Try_later (Client,User,Client is busy )")
    else
      if State =SC
      then Send ("Try_later(Client,user,All_Servers are busy )")
    end
  end
end
procedure open (Semaphore ,Client )
begin
  if State =SB
  then State;SC
    Send ("s_job (Client,Server,Document)")
  end
end
procedure Closed (Semaphore ,Client)
begin
  Countdown ;Waiting Period
  Send ("Wait(Client,Client)")
end

procedure Wait (client ,Client)
begin
  Countdown ;Countdown-1
  if Countdown>0
  then Send ("wait(Client,Client)")
  else Send ("Are_you_open(Client,SEMAPHORE (Semaphore Index))")
  end

procedure C_Ready (Server,Client)
begin
  State ;SA
end

```

Documentation 3

PRINT

Communication Structure Diagram 1D1/21

```

©©©©©©©©©©©©
©    USER ©©©©Ó»
©©©©©©©©©©©©
©      i
C_Job  Try_Later©©©©¹Ò»¶ÔÊ
      i      ©
©©©©©©©©©©©©
© S_Job ©  CLIENT  ©©©©¿»»(´´ÖCOMPUTER)

```

```

© © i ©©©©©©©©©©©©
© © i Are_you_open i
© © © ©
© © open © Closed
© © © i ©
© © ©©©©©©©©©©©©
© © © SEMAPHORE ©©©Ð°µ
© © ©©©©©©©©©©©©
© © i
© C_Ready Ready©©©¾Ð(Ò×±°Á)
© © ©
© ©©©©©©©©©©©©©
©©©©©i© SERVER ©© ·îÆ(¨Ö´Ó»)
©©©©©©©©©©©©
(i5.3-3)

```

[illegible]

Documentation 4, Answer Subtask a.3    PRINT    Competitor Code  
SEMAPHORE\$

© received © Stata/ © sent ©  
© messages © Transition © messages ©

```

time
  @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
0 @           @   S1           @           @
  @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1 @           @   S1           @           @
  @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
2 @Are_you_open @   S1;S2       @ open       @
  @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
3 @Are_you_open @   S2           @ closed     @
  @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
4 @           @   S2           @           @

```

CLIENT☐

time

CLIENT☐

time





[illegible]

