



This document is Undip Institutional Repository Collection. The author(s) or copyright owner(s) agree that UNDIP-IR may, without changing the content, translate the submission to any medium or format for the purpose of preservation. The author(s) or copyright owner(s) also agree that UNDIP-IR may keep more than one copy of this submission for purpose of security, back-up and preservation:

(<http://eprints.undip.ac.id>)

LAMPIRAN A

```
Program Solusi_Dekker;
uses Crt;
Const
  N = 2;
Var
  turn : integer;
  flag : array [0..N-1] of boolean;

Procedure MasukCriticalSection;
begin
  write ('0');
end;

Procedure KeluarCriticalSection;
begin
  write ('1');
end;

Procedure Proses0;
begin
repeat
  flag[0] := true;
  turn    := 1;
  while flag[1] do
    if turn = 1 then
      begin
        flag[0]    := false;
        while turn = 1 do
          flag[0]    := true;
      end;
    MasukCriticalSection;
    turn    := 1;
    flag[0] := false;
    KeluarCriticalSection;
  until false
end;
```

```
Procedure Proses1;
begin
repeat
    flag[1] := true;
    turn := 0;
    while flag[0] do
        if turn = 0 then
            begin
                flag[1] := false;
                while turn = 0 do
                    flag[1] := true;
            end;
        MasukCriticalSection;
        turn := 0;
        flag[1] := false;
        KeluarCriticalSection;
    until false
end;

Begin
    ClrScr;
    flag[0] := false;
    flag[1] := false;
    turn := 0;
begin
    Proses0;
    Proses1;
end;
readln;
End.
```

LAMPIRAN B

```
Program Solusi_Dekker;
uses Crt;
Const
    N = 2;
Var
    turn, I : integer;
    flag     : array [0..N-1] of boolean;

Procedure MasukCriticalSection;
begin
    write (' A',I);
end;

Procedure KeluarCriticalSection;
begin
    write (' B',I);
end;

Procedure ProsesA;
begin
    flag[0]    := true;
    while flag[1] do
        if turn = 1 then
            begin
                flag[0]    := false;
                while turn = 1 do
                    flag[0]    := true;
            end;
    MasukCriticalSection;
    turn    := 1;
    flag[0] := false;
    KeluarCriticalSection;
    I       := I+1;
end;
```

```
Procedure ProsesB;
begin
    flag[1] := true;
    while flag[0] do
        if turn = 0 then
            begin
                flag[1] := false;
                while turn = 0 do
                    flag[1] := true;
            end;
        repeat
            MasukCriticalSection;
            turn := 0;
            flag[1] := false;
            KeluarCriticalSection;
            I := I+1;
        until I = 222;
    end;

Begin
    ClrScr;
    flag[0] := false;
    flag[1] := false;
    turn := 0;
    I := 1;
    begin
        ProsesA;
        ProsesB;
    end;
    readln
End.
```

