323.25 Project 7: Scheduling

Student: Jingshi Liu

Due Date: 4/19/2022

Algorithm Steps:

```
Step 1:  inFile1, inFile2, outFile1, outFile2 ← open
         numNodes ← read from inFile1.
         numProcs ← get from argv [3]
         if (numProcs <= 0) exit with error message "need 1 or more processors".
                  else if (numProcs > numNodes)
                           numProcs ← numNodes // means unlimited processors.
Step 2:  Matrix ← dynamically allocate, size of numNodes+1 by numNodes+1, initialize to zero

         loadMatrix (inFile1)
         setMatrix (…)
         printMatrix(outFile2) // check for your self to make sure the matrix is correctly set.
Step 3:  OPEN ← get a dummy node for OPEN to point to
         currentTime ← 0 // at the beginning of scheduling
         procUsed ← 0 // no processor is used at the beginning
Step 4:  totalJobTimes ← loadJobTimeAry (inFile2)
         Table  ← dynamically allocate, size of numProcs by totalJobTimes, initialize to zero
         printTable (outFile1, currentTime)
Step 5: fillOPEN (…)
         printOPEN (outFile2)
Step 6: fillTable (…)
        printTable (outFile1, currentTime)
Step 7: currentTime ++
Step 8: deleteDoneJobs (…)
Step 9: if checkCycle (…) is true
                  output error message to outFile1: "there is cycle in the graph!!!" and exit the program
step 10: repeat step 3 to step 7 until isGraphEmpty (…)
step 11: printTable (outFile1) // The final schedule table.
step 12: close all files
```

Note. All the OutFiles have around thousand of lines in total, they will be included in the email to promote the readability.

Illustration for Graph1 with 3 processors

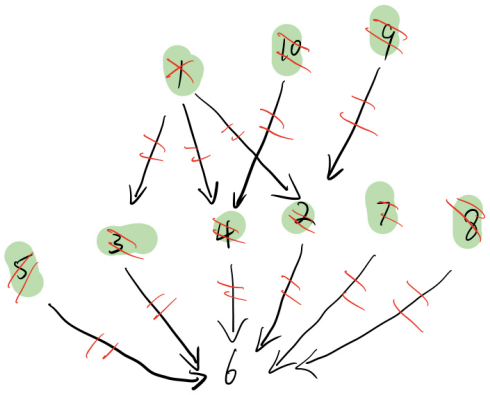


table	0	1	2	3	4	5	6	7	8
P 1	1	8	3	6					
P 2	5	9	2						
P 3	7	10	4						

open : 1, 5, 7, 8, 9, 10

↓

open : 8, 9, 10, 3

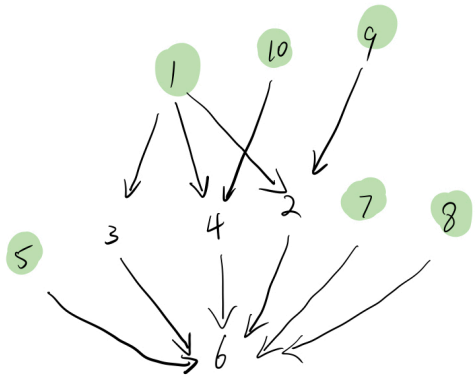
↓

open : 3, 2, 4

↓

open : 6

Illustration for graph 1 with infinite processors



tuble	0	1	2	3	4	5	6	7	8
p 1	1	2	6						
p 2	5	3							
p 3	7	4							
p 4	8								
p5	9								
p 6	10								

open 1, 5, 7, 8, 9, 10

↓

open 2, 3, 4

↓

open 6

Source Code:

```
import java.io.*;
import java.util.Scanner;

class Node{
    int jobID;
    int jobTime;
    Node next;

    public Node(int jobID, int jobTime, Node next) {
        this.jobID = jobID;
        this.jobTime = jobTime;
        this.next = next;
    }

    String getInfo(){
        return "("+ jobID + ", " + jobTime + ") --> ";
    }
}

class Scheduler{
    int numNodes, numProcs, procUsed, currentTime, totalJobTimes;
    int[] jobTimeAry;
    int[][] matrix, table;
    Node OPEN;

    public Scheduler(int numNodes, int numProcs) {
        this.numNodes = numNodes;
        this.numProcs = numProcs;
```

```
    jobTimeAry = new int[numNodes+1];
    matrix = new int[numNodes+1][numNodes+1];
    OPEN = new Node(0, 0, null);
}

void allocateTable(){
    table = new int[numProcs+1][this.totalJobTimes+1];
}

void loadMatrix(Scanner graph){
    int node1, node2;
    while(graph.hasNext()){
        node1 = graph.nextInt();
        node2 = graph.nextInt();
        matrix[node1][node2] = 1;

        // following lines of code increment the parent count and dependent count
        matrix[node1][0]++;
        matrix[0][node2]++;
    }
    // here initialize the state of the node: matrix[i][i] = 1
    matrix[0][0] = numNodes;
    for (int i = 1; i < numNodes+1; i++) {
        matrix[i][i] = 1;
    }
}

int loadJobTimeAry(Scanner inFile){
    inFile.nextInt();
    int node, time;
    while(inFile.hasNext()){
        node = inFile.nextInt();
```

```
        time = inFile.nextInt();
        jobTimeAry[node] = time;
        totalJobTimes += time;
    }
    return totalJobTimes;
}

void printMatrix(BufferedWriter outFile) throws IOException {
    outFile.write("\n\n\n\nMatrix:\n        ");
    for (int i = 0; i < matrix.length; i++) {
        if(i!=0){
            outFile.write("Node " + i + " :" );
        }

        for (int j = 0; j < matrix.length; j++) {
            outFile.write(" " + matrix[i][j] + " ");
        }
        outFile.write("\n");
    }
}

int findOrphan(){
    for (int i = 0; i < matrix.length; i++) {
        if(matrix[0][i] == 0 && matrix[i][i] == 1){
            matrix[i][i] = 2;
            return i;
        }
    }
    return -1;
}

void openInsert(Node node){
```

```
    int val = node.jobTime;
    Node spot = OPEN;

    //if use spot.next.jobTime < val as condition, a new job has same job time
    // compared to a JOB in open can cut the line in front of JOB
    while(spot.next != null && spot.next.jobTime <= val){
        spot = spot.next;
    }

    node.next = spot.next;
    spot.next = node;
}

// format: (jobID, jobTime) -->
void printOpen(BufferedWriter outFile) throws IOException {
    Node pointer = OPEN;
    outFile.write("\nOPEN: ");
    while(pointer != null){
        outFile.write(pointer.getInfo());
        pointer = pointer.next;
    }
    outFile.write("null");
}

int getNextProc(){
    for (int i = 1; i < table.length; i++) {
        if(table[i][currentTime] == 0) return i;
    }
    return -1;
}

void fillOpen(){
```

```
    int jobID = findOrphan();
    while(jobID != -1){
        openInsert(new Node(jobID, jobTimeAry[jobID], null));
        jobID = findOrphan();
    }
}

void fillTable(){
    int availableProc = getNextProc();
    Node newJob;
    while(availableProc >= 0 && OPEN.next != null && procUsed <= numProcs){
        newJob = removeOpen();
        putJobOnTable(availableProc, newJob);
        if(availableProc > procUsed)
            procUsed++;
        availableProc = getNextProc();
    }
}

void putJobOnTable(int availableProc, Node newJob) {
    int endTime = currentTime + newJob.jobTime;
    for (int time = currentTime; time < endTime; time++) {
        table[availableProc][time] = newJob.jobID;
    }
}

Node removeOpen(){
    Node node = OPEN.next;
    OPEN.next = OPEN.next.next;
    node.next = null;
    return node;
}
```

```
void printTable(BufferedWriter outFile) throws IOException{
    outFile.write("\n\n\n\n==================================================\n\n" +
            "ProcUsed: " + procUsed + "     currentTime: " + currentTime +
            "\n\nTime:  ");
    for (int i = 0; i < totalJobTimes; i++) {
        if(i < 10)
            outFile.write(" |  " + i);
        else
            outFile.write(" | " + i);
    }

    for (int procNumber = 1; procNumber <= procUsed; procNumber++) {
        outFile.write("\n\nProc: " + procNumber);
        for (int time = 0; time < totalJobTimes; time++) {
            if(table[procNumber][time] == 0)
                outFile.write(" | --");
            else if(table[procNumber][time] < 10)
                outFile.write(" |  " + table[procNumber][time]);
            else
                outFile.write(" | " + table[procNumber][time]);
        }
    }
}

boolean containsCycle(){
    boolean allProcsResting = true;
    for (int i = 1; i <= procUsed; i++) {
        if(table[i][currentTime] > 0) {
            allProcsResting = false;
            break;
        }
```

```
        }

        return allProcsResting && OPEN.next == null && (!graphIsEmpty());
    }

    boolean graphIsEmpty(){
        return matrix[0][0] == 0;
    }

    void deleteDoneJobs(){
        for (int proc = 1; proc <= procUsed; proc++) {
            if(table[proc][currentTime] == 0 && table[proc][currentTime-1] != 0) {
                deleteJob(table[proc][currentTime - 1]);
            }
        }
    }

    void deleteJob(int jobID){
        matrix[jobID][jobID] = 0;
        matrix[0][0]--;
        matrix[jobID][0] = 0;
        for (int i = 1; i <= numNodes; i++) {
            if(matrix[jobID][i] > 0){
                matrix[jobID][i] = 0;
                matrix[0][i]--;
            }
        }
    }
}

public class Main {
```

```
public static void main(String[] args) throws IOException {
    Scanner graph, jobTime;
    BufferedWriter outFile, debugFile;
    try{
        graph = new Scanner(new File(args[0]));
        jobTime = new Scanner(new File(args[1]));
        outFile = new BufferedWriter(new FileWriter(args[2], true));
        debugFile = new BufferedWriter(new FileWriter(args[3], true));
    }catch (IOException exception){
        return;
    }

    int numNodes = graph.nextInt();
    int numProcs = new Scanner(args[4]).nextInt();

    try{
        if (numProcs <= 0)
            throw new Exception("Invalid input from args[4] for numProcs.");
    }catch (Exception ignored){}

    // Constructor and other methods to fully initialize the Scheduler
    Scheduler scheduler = new Scheduler(numNodes,numProcs);
    scheduler.loadMatrix(graph);
    scheduler.printMatrix(debugFile);
    scheduler.loadJobTimeAry(jobTime);
    scheduler.allocateTable();
    scheduler.printTable(outFile);


    while(!scheduler.graphIsEmpty()){
        scheduler.fillOpen();
        scheduler.printOpen(debugFile);
```

```
            // Has to check after fillOpen
            // Otherwise, nodes that might not make a cycle are not yet pushed into OPEN
            if(scheduler.containsCycle()) {
                outFile.write("\n\nFound a cycle in the graph!!!");
                break;
            }

            scheduler.fillTable();
            scheduler.printTable(outFile);

            scheduler.currentTime++;
            scheduler.deleteDoneJobs();

        }
        scheduler.printTable(outFile);
        scheduler.printMatrix(debugFile);

        graph.close();
        jobTime.close();
        outFile.close();
        debugFile.close();
    }
}
```

OutFiles:

3)

outFile:

==========

ProcUsed: 0 currentTime: 0

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Proc: 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 0

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Proc: 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 5 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 7 | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3   currentTime: 1

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Proc: 1 | 1 | 8 | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 5 | 9 | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 7 | 10 | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3   currentTime: 2

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Proc: 1 | 1 | 8 | 3 | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 5 | 9 | 2 | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 7 | 10 | 4 | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3   currentTime: 3

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Proc: 1 | 1 | 8 | 3 | 6 | -- | -- | -- | -- | -- | --

Proc: 2 | 5 | 9 | 2 | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 7 | 10 | 4 | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 4

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Proc: 1 | 1 | 8 | 3 | 6 | -- | -- | -- | -- | -- | --

Proc: 2 | 5 | 9 | 2 | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 7 | 10 | 4 | -- | -- | -- | -- | -- | -- | —

Debugging:

Matrix:

	10	0	2	1	2	0	6	0	0	0	0
Node 1 :	3	1	1	1	1	0	0	0	0	0	0
Node 2 :	1	0	1	0	0	0	1	0	0	0	0
Node 3 :	1	0	0	1	0	0	1	0	0	0	0
Node 4 :	1	0	0	0	1	0	1	0	0	0	0
Node 5 :	1	0	0	0	0	1	1	0	0	0	0
Node 6 :	0	0	0	0	0	0	1	0	0	0	0
Node 7 :	1	0	0	0	0	0	1	1	0	0	0

Node 8 : 1 0 0 0 0 0 1 0 1 0 0

Node 9 : 1 0 1 0 0 0 0 0 0 1 0

Node 10 : 1 0 0 0 1 0 0 0 0 0 1

OPEN: (0, 0) --> (1, 1) --> (5, 1) --> (7, 1) --> (8, 1) --> (9, 1) --> (10, 1) --> null

OPEN: (0, 0) --> (8, 1) --> (9, 1) --> (10, 1) --> (3, 1) --> null

OPEN: (0, 0) --> (3, 1) --> (2, 1) --> (4, 1) --> null

OPEN: (0, 0) --> (6, 1) --> null

Matrix:

0 0 0 0 0 0 0 0 0 0 0

Node 1 : 0 0 0 0 0 0 0 0 0 0 0

Node 2 : 0 0 0 0 0 0 0 0 0 0 0

Node 3 : 0 0 0 0 0 0 0 0 0 0 0

Node 4 : 0 0 0 0 0 0 0 0 0 0 0

Node 5 : 0 0 0 0 0 0 0 0 0 0 0

Node 6 : 0 0 0 0 0 0 0 0 0 0 0

Node 7 : 0 0 0 0 0 0 0 0 0 0 0

Node 8 : 0 0 0 0 0 0 0 0 0 0 0

Node 9 : 0 0 0 0 0 0 0 0 0 0 0

Node 10 : 0 0 0 0 0 0 0 0 0 0 0

5) Graph1 unlimited processors

==========

ProcUsed: 0 currentTime: 0

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

==========

ProcUsed: 6 currentTime: 0

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Proc: 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 5 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 7 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 8 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 9 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 10 | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 6   currentTime: 1

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Proc: 1 | 1 | 2 | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 5 | 3 | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 7 | 4 | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 8 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 9 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 10 | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 6 currentTime: 2

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Proc: 1 | 1 | 2 | 6 | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 5 | 3 | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 7 | 4 | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 8 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 9 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 10 | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 6 currentTime: 3

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Proc: 1 | 1 | 2 | 6 | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 5 | 3 | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 7 | 4 | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 8 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 9 | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 10 | -- | -- | -- | -- | -- | -- | -- | -- | —

Debugging:

Matrix:

10 0 2 1 2 0 6 0 0 0 0

Node 1 : 3 1 1 1 1 0 0 0 0 0 0

Node 2 : 1 0 1 0 0 0 1 0 0 0 0

Node 3 : 1 0 0 1 0 0 1 0 0 0 0

Node 4 : 1 0 0 0 1 0 1 0 0 0 0

Node 5 : 1 0 0 0 0 1 1 0 0 0 0

Node 6 : 0 0 0 0 0 0 1 0 0 0 0

Node 7 : 1 0 0 0 0 0 1 1 0 0 0

Node 8 : 1 0 0 0 0 0 1 0 1 0 0

Node 9 : 1 0 1 0 0 0 0 0 0 1 0

Node 10 : 1 0 0 0 1 0 0 0 0 0 1

OPEN: (0, 0) --> (1, 1) --> (5, 1) --> (7, 1) --> (8, 1) --> (9, 1) --> (10, 1) --> null

OPEN: (0, 0) --> (2, 1) --> (3, 1) --> (4, 1) --> null

OPEN: (0, 0) --> (6, 1) --> null

Matrix:

0 0 0 0 0 0 0 0 0 0 0

Node 1 : 0 0 0 0 0 0 0 0 0 0 0

Node 2 : 0 0 0 0 0 0 0 0 0 0 0

Node 3 : 0 0 0 0 0 0 0 0 0 0 0

Node 4 : 0 0 0 0 0 0 0 0 0 0 0

Node 5 : 0 0 0 0 0 0 0 0 0 0 0

Node 6 : 0 0 0 0 0 0 0 0 0 0 0

Node 7 : 0 0 0 0 0 0 0 0 0 0 0

Node 8 : 0 0 0 0 0 0 0 0 0 0 0

Node 9 : 0 0 0 0 0 0 0 0 0 0 0

Node 10 : 0 0 0 0 0 0 0 0 0 0 0

6) Set 3 , 3 processors

Outfile:

==========

ProcUsed: 0 currentTime: 0

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26

==========

ProcUsed: 3 currentTime: 0

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26

Proc: 1 | 10 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | --

Proc: 3 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | --

===

ProcUsed: 3   currentTime: 1

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
20 | 21 | 22 | 23 | 24 | 25 | 26

Proc: 1 | 10 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | --

Proc: 2 | 12 | 13 | 13 | 13 | 13 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 3 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | --

===

ProcUsed: 3 currentTime: 2

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26

Proc: 1 | 10 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 12 | 13 | 13 | 13 | 13 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 1 | 1 | 3 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 3

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26

Proc: 1 | 10 | 11 | 11 | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 12 | 13 | 13 | 13 | 13 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 1 | 1 | 3 | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 4

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26

Proc: 1 | 10 | 11 | 11 | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 12 | 13 | 13 | 13 | 13 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 1 | 1 | 3 | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 5

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26

Proc: 1 | 10 | 11 | 11 | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 12 | 13 | 13 | 13 | 13 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 1 | 1 | 3 | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Found a cycle in the graph!!!

==========

ProcUsed: 3 currentTime: 6

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26

Proc: 1 | 10 | 11 | 11 | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 12 | 13 | 13 | 13 | 13 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 1 | 1 | 3 | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | —

Debugging:

Matrix:

	14	0	1	1	1	1	5	5	4	1	0	0	0	0	2
Node 1 :	3	1	1	1	1	0	0	0	0	0	0	0	0	0	0
Node 2 :	2	0	1	0	0	0	1	1	0	0	0	0	0	0	0
Node 3 :	2	0	0	1	0	0	1	1	0	0	0	0	0	0	0
Node 4 :	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0
Node 5 :	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0
Node 6 :	2	0	0	0	0	0	1	0	1	0	0	0	0	0	1
Node 7 :	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0
Node 8 :	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0
Node 9 :	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0
Node 10 :	2	0	0	0	0	0	1	0	0	0	1	0	0	0	1
Node 11 :	2	0	0	0	0	0	0	1	1	0	0	1	0	0	0
Node 12 :	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0
Node 13 :	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0
Node 14 :	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1

OPEN: (0, 0) --> (10, 1) --> (12, 1) --> (1, 2) --> (11, 2) --> (13, 4) --> null

OPEN: (0, 0) --> (11, 2) --> (13, 4) --> null

OPEN: (0, 0) --> (3, 1) --> (4, 2) --> (2, 3) --> null

OPEN: (0, 0) --> (4, 2) --> (2, 3) --> null

OPEN: (0, 0) --> null

OPEN: (0, 0) --> null

OPEN: (0, 0) --> null

Matrix:

6 0 0 0 0 1 1 1 2 1 0 0 0 0 1

Node 1 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 2 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 3 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 4 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 5 : 1 0 0 0 0 1 1 0 0 0 0 0 0 0 0

Node 6 : 2 0 0 0 0 0 1 0 1 0 0 0 0 0 1

Node 7 : 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0

Node 8 : 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0

Node 9 : 1 0 0 0 0 0 0 0 1 1 0 0 0 0 0

Node 10 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 11 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 12 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 13 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 14 : 1 0 0 0 0 1 0 0 0 0 0 0 0 0 1

7) Set 4 with 3 processors

Outfile:

==

ProcUsed: 0   currentTime: 0

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

==

ProcUsed: 3   currentTime: 0

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

===

ProcUsed: 3   currentTime: 1

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

===

ProcUsed: 3 currentTime: 2

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 3

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 4

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 5

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 6

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 7

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | 2 | 2 | 2 | 7 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | 6 | 6 | 6 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 8

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | 2 | 2 | 2 | 7 | 9 | 9 | 9 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | 6 | 6 | 6 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 9

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | 2 | 2 | 2 | 7 | 9 | 9 | 9 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | 6 | 6 | 6 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 10

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | 2 | 2 | 2 | 7 | 9 | 9 | 9 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | 6 | 6 | 6 | 14 | 14 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 11

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | 2 | 2 | 2 | 7 | 9 | 9 | 9 | 8 | 8 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | 6 | 6 | 6 | 14 | 14 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3   currentTime: 12

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | 2 | 2 | 2 | 7 | 9 | 9 | 9 | 8 | 8 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | 6 | 6 | 6 | 14 | 14 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 3 currentTime: 13

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | 1 | 1 | 3 | 2 | 2 | 2 | 7 | 9 | 9 | 9 | 8 | 8 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | 12 | 12 | 4 | 4 | -- | -- | 6 | 6 | 6 | 14 | 14 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | —

Debugging:

Matrix:

	14	0	1	1	1	0	5	4	4	1	0	0	0	0	2
Node 1 :	3	1	1	1	1	0	0	0	0	0	0	0	0	0	0
Node 2 :	2	0	1	0	0	0	1	1	0	0	0	0	0	0	0
Node 3 :	2	0	0	1	0	0	1	1	0	0	0	0	0	0	0
Node 4 :	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0

Node 5 : 1 0 0 0 0 1 1 0 0 0 0 0 0 0 0

Node 6 : 2 0 0 0 0 0 1 0 1 0 0 0 0 0 1

Node 7 : 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0

Node 8 : 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0

Node 9 : 1 0 0 0 0 0 0 0 1 1 0 0 0 0 0

Node 10 : 2 0 0 0 0 0 1 0 0 0 1 0 0 0 1

Node 11 : 2 0 0 0 0 0 0 1 1 0 0 1 0 0 0

Node 12 : 1 0 0 0 0 0 0 0 1 0 0 0 1 0 0

Node 13 : 1 0 0 0 0 0 1 0 0 0 0 0 0 1 0

Node 14 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

OPEN: (0, 0) --> (5, 1) --> (10, 1) --> (13, 1) --> (1, 2) --> (12, 2) --> (11, 4) --> null

OPEN: (0, 0) --> (1, 2) --> (12, 2) --> (11, 4) --> null

OPEN: (0, 0) --> null

OPEN: (0, 0) --> (3, 1) --> (4, 2) --> (2, 3) --> null

OPEN: (0, 0) --> (2, 3) --> null

OPEN: (0, 0) --> null

OPEN: (0, 0) --> null

OPEN: (0, 0) --> (7, 1) --> (6, 3) --> null

OPEN: (0, 0) --> (9, 3) --> null

OPEN: (0, 0) --> null

OPEN: (0, 0) --> (14, 2) --> null

OPEN: (0, 0) --> (8, 2) --> null

OPEN: (0, 0) --> null

Matrix:

0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 1 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 2 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 3 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 4 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 5 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 6 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 7 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 8 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 9 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 10 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 11 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 12 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 13 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 14 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0

8) Set 4 with unlimited processors

Out file:

===

ProcUsed: 0   currentTime: 0

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

===

ProcUsed: 6   currentTime: 0

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --
| -- | -- | -- | -- | -- | --

===

ProcUsed: 6   currentTime: 1

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==

ProcUsed: 6   currentTime: 2

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | 3 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==

ProcUsed: 6   currentTime: 3

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | 3 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==

ProcUsed: 6   currentTime: 4

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | 3 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | 4 | 4 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==

ProcUsed: 6   currentTime: 5

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | 3 | -- | -- | 7 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | 4 | 4 | -- | 6 | 6 | 6 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 6 currentTime: 6

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | 3 | -- | -- | 7 | 9 | 9 | 9 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | 4 | 4 | -- | 6 | 6 | 6 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 6 currentTime: 7

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | 3 | -- | -- | 7 | 9 | 9 | 9 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | 4 | 4 | -- | 6 | 6 | 6 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --
| -- | -- | -- | -- | -- | --

==

ProcUsed: 6   currentTime: 8

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | 3 | -- | -- | 7 | 9 | 9 | 9 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | 4 | 4 | -- | 6 | 6 | 6 | 14 | 14 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --
| -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- |
-- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==

ProcUsed: 6   currentTime: 9

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | 3 | -- | -- | 7 | 9 | 9 | 9 | 8 | 8 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | 4 | 4 | -- | 6 | 6 | 6 | 14 | 14 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 6 currentTime: 10

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | 3 | -- | -- | 7 | 9 | 9 | 9 | 8 | 8 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | 4 | 4 | -- | 6 | 6 | 6 | 14 | 14 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

==========

ProcUsed: 6 currentTime: 11

Time: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27

Proc: 1 | 5 | -- | 3 | -- | -- | 7 | 9 | 9 | 9 | 8 | 8 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 2 | 10 | -- | 4 | 4 | -- | 6 | 6 | 6 | 14 | 14 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 3 | 13 | -- | 2 | 2 | 2 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 4 | 1 | 1 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 5 | 12 | 12 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | --

Proc: 6 | 11 | 11 | 11 | 11 | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | —

Debugging:

Matrix:

14 0 1 1 1 0 5 4 4 1 0 0 0 0 2

Node 1 : 3 1 1 1 1 0 0 0 0 0 0 0 0 0 0

Node 2 : 2 0 1 0 0 0 1 1 0 0 0 0 0 0 0

Node 3 : 2 0 0 1 0 0 1 1 0 0 0 0 0 0 0

Node 4 : 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0

Node 5 : 1 0 0 0 0 1 1 0 0 0 0 0 0 0 0

Node 6 : 2 0 0 0 0 0 1 0 1 0 0 0 0 0 1

Node 7 : 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0

Node 8 : 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0

Node 9 : 1 0 0 0 0 0 0 0 1 1 0 0 0 0 0

Node 10 : 2 0 0 0 0 0 1 0 0 0 1 0 0 0 1

Node 11 : 2 0 0 0 0 0 0 1 1 0 0 1 0 0 0

Node 12 : 1 0 0 0 0 0 0 0 1 0 0 0 1 0 0

Node 13 : 1 0 0 0 0 0 1 0 0 0 0 0 0 1 0

Node 14 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

OPEN: (0, 0) --> (5, 1) --> (10, 1) --> (13, 1) --> (1, 2) --> (12, 2) --> (11, 4) --> null

OPEN: (0, 0) --> null

OPEN: (0, 0) --> (3, 1) --> (4, 2) --> (2, 3) --> null

OPEN: (0, 0) --> null

OPEN: (0, 0) --> null

OPEN: (0, 0) --> (7, 1) --> (6, 3) --> null

OPEN: (0, 0) --> (9, 3) --> null

OPEN: (0, 0) --> null

OPEN: (0, 0) --> (14, 2) --> null

OPEN: (0, 0) --> (8, 2) --> null

OPEN: (0, 0) --> null

Matrix:

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 1 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 2 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 3 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 4 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 5 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 6 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 7 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 8 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 9 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 10 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 11 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 12 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 13 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Node 14 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0