

Cover Page

Name: David Chen Salas

Section: 2023 Fall Term (1) Algorithms I CSCI 700 231[25504] (Queens College)

Project#: 6

Project Name: Partial ordering, dependency graph and Scheduling

Due Date: 11/17/2023 Friday before midnight

Algorithm Steps:

Step 1: inFile1, inFile2, outFile1, debugFile open via args []
numNodes read from inFile1.
numProcs get from args [2]
if (numProcs <= 0)
console write error message "need 1 or more processors".
exit the program
Else
if (numProcs > numNodes)
numProcs = numNodes // means unlimited processors.

Step 2: Matrix dynamically allocate, size of numNodes+1 by numNodes+1
loadMatrix (inFile1, Matrix)
outFile1 "In main () after loadMatrix ()"
printMatrix (outFile1) // check for yourself to make sure the matrix is loaded correctly.
setMatrix (Matrix, debugFile)
outFile1 "In main () after setMatrix ()"
printMatrix (outFile1) // check for yourself to make sure the matrix is correctly set.

Step 3: Open get a ListNode (-99, 0, null) as 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, jobTimeAry, debugFile)
Table dynamically allocate, size of numProcs by totalJobTimes, initialize to zero.
outFile1 "in main() after allocate and initialize table"
printTable (outFile1, currentTime)

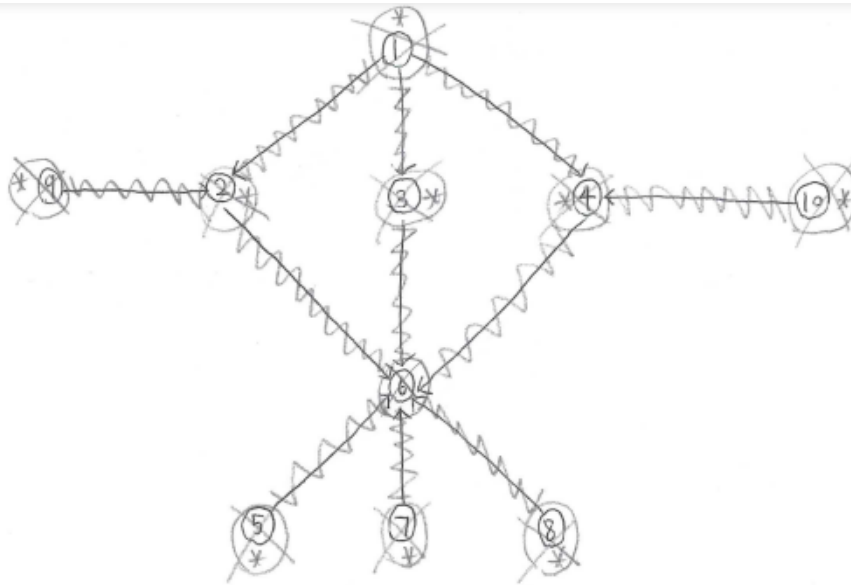
Step 5: scheduling (outFile1, debugFile)

Step 6: outFile1 "in main() printing Table and Open"
printTable (outFile1, currentTime) // The final schedule table.
printOpen (outFile1, currentTime)

Step 7: close all files

Illustration

****Hand Draw Set1 with 3 processors****



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	-						

open(0): ~~1~~ → ~~5~~ → ~~2~~ → 8 → 9 → 10

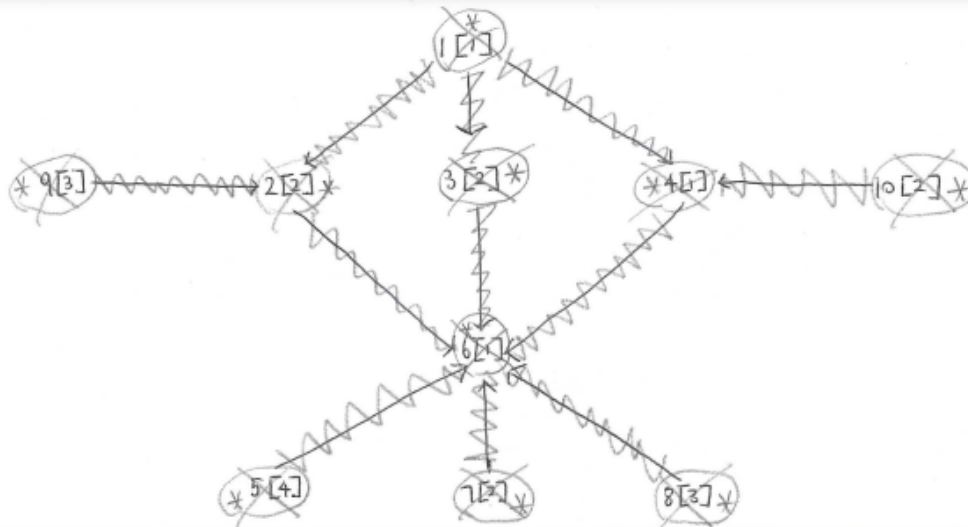
open(1): ~~8~~ → ~~2~~ → ~~10~~ → 3

open(2): ~~3~~ → ~~2~~ → ~~4~~

open(3): ~~8~~

open(4): empty

****Hand Draw Set2 with unlimited processors****



Time:	0	1	2	3	4	5	6	7	8	9
Proc#1:	1	3	3	2	2	6				
Proc#2:	7	7	4	-	-	-				
Proc#3:	10	10	-	-	-	-				
Proc#4:	8	8	8	-	-	-				
Proc#5:	9	9	9	-	-	-				
Proc#6:	5	5	5	5	-	-				

open(0): $(1,1) \rightarrow (7,2) \rightarrow (10,2) \rightarrow (8,3) \rightarrow (9,3) \rightarrow (5,4)$

open(1): $(3,2)$

open(2): $(4,1)$

open(3): $(2,2)$

open(4): —

open(5): $(6,1)$

open(6): empty

Source Code

```
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Scanner;

public class ChenSalasD_Project6_Main {

    static Scanner inFile1, inFile2;
    static int numProcessors;
    static FileWriter outFile1, debugFile;

    static class listNode{
        int jobID;
        int jobTime;
        listNode next;

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

    static int numNodes;
    static int numProcs, procUsed;
    static int currentTime = 0;
    static int totalJobTimes;
    static int[] jobTimeAry;

    static int[][] Matrix;
    static int[][] Table;

    static listNode Open;

    public static void main(String[] args) throws IOException {
        inFile1 = new Scanner(new FileReader(args[0]));
        inFile2 = new Scanner(new FileReader(args[1]));
        numProcessors = Integer.parseInt(args[2]);
        outFile1 = new FileWriter(args[3]);
        debugFile = new FileWriter(args[4]);

        numNodes = inFile1.nextInt();
        numProcs = numProcessors;
        if(numProcs <= 0){
            System.out.println("Need 1 or more processors");
            System.exit(0);
        }
        if(numProcs > numNodes){
            numProcs = numNodes;
        }

        //Step 2
        Matrix = new int[numNodes+1][numNodes+1];
        loadMatrix(inFile1, Matrix);
        outFile1.write("In main() after loadMatrix().\n");
        printMatrix(outFile1);
        setMatrix(Matrix, debugFile);
        outFile1.write("\nIn main() after setMatrix().\n");
        printMatrix(outFile1);
    }
}
```

```

//Step 3
Open = new listNode(-99, 0, null);
currentTime = 0;
procUsed = 0;

//Step 4
jobTimeAry = new int[numNodes + 1];
totalJobTimes = loadJobTimeAry(inFile2, jobTimeAry, debugFile);
Table = new int[numProcs + 1][totalJobTimes + 1];
outFile1.write( "\nin main() after allocate and initialize table.\n");
printTable(outFile1, currentTime);

//Step 5
scheduling(outFile1, debugFile);

//Step 6
outFile1.write("In main() printing Table and Open\n");
printTable(outFile1, currentTime);
printOpen(outFile1, currentTime);

inFile1.close();
inFile2.close();
outFile1.close();
debugFile.close();
}

public static void loadMatrix(Scanner inFile1, int[][] matrix) {
    while(inFile1.hasNext()){
        matrix[inFile1.nextInt()][inFile1.nextInt()] = 1;
    }
}

public static void printMatrix(FileWriter outFile) throws IOException {
    for(int i = 0; i < numNodes + 1; i++){
        for (int j = 0; j < numNodes + 1; j++) {
            outFile.write(Matrix[i][j] + " ");
        }
        outFile.write("\n");
    }
}

public static void setMatrix(int[][] matrix, FileWriter debugFile) throws IOException {
    debugFile.write("Entering setMatrix method!\n");
    Matrix[0][0] = numNodes;
    for (int j = 1; j <= numNodes; j++) {
        matrix[0][j] = countParent(matrix, j, debugFile);
        matrix[j][0] = countDependent(matrix, j, debugFile);
    }
    for (int i = 1; i <= numNodes; i++) {
        matrix[i][i] = 1;
    }
    debugFile.write("Before leaving setMatrix method, the Matrix below\n");
    printMatrix(debugFile);
    debugFile.write("Leaving setMatrix().\n");
}

public static int countParent(int[][] matrix, int j, FileWriter debugFile) throws IOException {
    debugFile.write("Entering countParent() method!\n");
    int count = 0;
    for(int i = 1; i <= numNodes; i++){
        if(matrix[i][j] > 0){

```

```

        count++;
    }
}
debugFile.write("In countParent() parent count is " + count);
debugFile.write("\nLeaving countParent() method.\n");
return count;
}

public static int countDependent(int[][] matrix, int j, FileWriter debugFile) throws IOException {
    debugFile.write("Entering countDependent() method!\n");
    int count = 0;
    for(int i = 1; i <= numNodes; i++){
        if(matrix[j][i] > 0){
            count++;
        }
    }
    debugFile.write("In countDependent() dependent count is " + count);
    debugFile.write("\nLeaving countDependent() method.\n");
    return count;
}

public static int loadJobTimeAry(Scanner inFile, int[] jobTimeAry, FileWriter debugFile) throws IOException {
    debugFile.write("Entering loadJobTimeAry() method!\n");
    int jobTime = 0;
    int num = inFile.nextInt();
    for(int i = 1; i <= num; i++){
        jobTimeAry[inFile.nextInt()] = inFile.nextInt();
        jobTime += jobTimeAry[i];
        debugFile.write(jobTime + "\n");
    }
    debugFile.write("Leaving loadJobTimeAry() method.\n");
    return jobTime;
}

public static void printTable(FileWriter outFile, int currentTime) throws IOException {
    for(int n = 0; n < 15+totalJobTimes*8; n++) {
        outFile.write("=");
    }

    outFile.write("\nProcUsed:" + procUsed + "  currentTime:" + currentTime + "\n");
    outFile.write("Time:\t ");
    for(int i = 0; i <= totalJobTimes; i++) {
        outFile.write("  " + i + "\t");
    }

    outFile.write("\n");
    for(int n = 0; n < 15+totalJobTimes*8; n++) {
        outFile.write("-");
    }
    outFile.write("\n");

    for(int i = 1; i <= numProcs; i++){
        outFile.write("Proc:" + i + "\t");
        for(int j = 0; j <= currentTime; j++){
            outFile.write("  " + Table[i][j] + "\t");
        }
    }

    outFile.write("\n");
    for(int n = 0; n < 15+totalJobTimes*8; n++) {
        outFile.write("-");
    }
    outFile.write("\n");
}

```

```

    }
    printOpen(outFile, currentTime);
}

public static void printOpen(FileWriter outFile, int currentTime) throws IOException {
    listNode tmp = Open.next;
    outFile.write("Open(Time_ " + currentTime + "): ");
    while(tmp!=null){
        outFile.write("(" + tmp.jobID + ", " + tmp.jobTime + ")");
        if(tmp.next!=null){
            outFile.write(" --> ");
        }
        tmp = tmp.next;
    }
    outFile.write("\n");
}

public static void scheduling(FileWriter outFile1, FileWriter deBugFile) throws IOException {
    deBugFile.write("Entering scheduling() method!\n");

    while(!isGraphEmpty()) {
        loadOpen();
        outFile1.write("In scheduling() after loadOpen currentTime is " + currentTime + "\n");
        printOpen(outFile1, currentTime);

        loadTable();
        outFile1.write("after loadTable currentTime is " + currentTime + "\n");
        printTable(outFile1, currentTime);

        currentTime++;
        printOpen(outFile1, currentTime);

        deleteDoneJobs(currentTime, deBugFile);

        if (checkCycle()) {
            outFile1.write("\n***There is cycle in the graph!!!***\n\n");
            System.exit(0);
            return;
        }
    }
    deBugFile.write("Leaving scheduling method.\n");
}

public static void loadOpen() throws IOException {
    listNode spot;
    deBugFile.write("Entering loadOpen() method!\n");
    for(int j = 1; j <= numNodes; j++){
        if(Matrix[0][j] == 0 && Matrix[j][j] == 1){
            Matrix[j][j] = 2;
            deBugFile.write("Find an orphan, the orphan is " + j + "\n");
            listNode newNode = new listNode(j, jobTimeAry[j], null);
            spot = findSpot(newNode);
            insertOneNode(spot, newNode);
        }
    }
    deBugFile.write("Before leaving loadOpen().\n");
    printOpen(deBugFile, currentTime);
    deBugFile.write("Leaving loadOpen() method.\n");
}

public static listNode findSpot(listNode newNode) {

```

```

        listNode spot = Open;
        while(spot.next != null && spot.next.jobTime < newNode.jobTime){
            spot = spot.next;
        }
        return spot;
    }

    public static void insertOneNode(listNode spot, listNode newNode) {
        newNode.next = spot.next;
        spot.next = newNode;
    }

    public static listNode popNode(listNode open){
        listNode tmp = open.next;
        open.next = tmp.next;
        tmp.next = null;
        return tmp;
    }

    public static void loadTable() throws IOException{
        debugFile.write("Entering loadTable() method!\n");
        printOpen(debugFile, currentTime);
        int availProc = getNextProc(currentTime, debugFile);
        while(availProc >= 0 && Open.next!=null && procUsed <= numProcs){
            listNode newNode = popNode(Open);
            putJobOnTable(availProc, currentTime, newNode.jobID, newNode.jobTime);
            if(availProc > procUsed){
                procUsed++;
            }
            availProc = getNextProc(currentTime, debugFile);
        }
        debugFile.write("Before leaving loadTable().\n");
        printTable(debugFile, currentTime);
        printOpen(debugFile, currentTime);
        debugFile.write("Leaving loadTable() method.\n");
    }

    public static int getNextProc(int currentTime, FileWriter debugFile) throws IOException {
        debugFile.write("Entering getNextProc() method!\n");
        int i = 1;
        boolean notFound = true;
        while(notFound && i <= numNodes && i <= numProcs){
            if(Table[i][currentTime] == 0){
                debugFile.write("in getNextProc() found an available proc#" + i + "\n");
                notFound = false;
                return i;
            }
            i++;
        }
        debugFile.write("Leaving getNextProc() method. No available proc can be found.\n");
        return -1;
    }

    public static void putJobOnTable(int availProc, int currentTime, int jobID, int jobTime) {
        int Time = currentTime;
        int EndTime = Time + jobTime;
        while(Time < EndTime){
            Table[availProc][Time] = jobID;
            Time++;
        }
    }

    public static void deleteDoneJobs(int currentTime, FileWriter debugFile) throws IOException {

```



```

    debugFile.write("Entering deleteDoneJobs() method!\n");
    int proc = 1;
    while(proc <= procUsed) {
        if (Table[proc][currentTime] <= 0 && Table[proc][currentTime - 1] > 0) {
            int jobID = Table[proc][currentTime - 1];
            deleteOneJob(jobID, debugFile);
        }
        proc++;
    }
    debugFile.write("Leaving deleteDoneJobs() method below is the updated matrix.\n");
    printMatrix(debugFile);
}

public static void deleteOneJob(int jobID, FileWriter debugFile) throws IOException {
    debugFile.write("Entering deleteOneJob() method!\n");

    Matrix[jobID][jobID] = 0;
    Matrix[0][0]--;
    for (int j = 1; j <= numNodes; j++) {
        if(Matrix[jobID][j] > 0){
            Matrix[0][j]--;
        }
    }
    debugFile.write("Leaving deleteOneJob() method.\n");
}

public static boolean checkCycle() {
    if(Open.next == null && Matrix[0][0] > 0){
        for(int i = 1; i <= numProcs; i++){
            if(Table[i][currentTime-1] > 0){
                return false;
            }
        }
        return true;
    }
    return false;
}

public static boolean isGraphEmpty() {
    if(Matrix[0][0] == 0){
        return true;
    }
    return false;
}
}

```

Program Output

*****Set_1 with 3 processors*****

****OutFile1****

In main() after loadMatrix().

```
0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0 0
```

In main() after setMatrix().

```
10 0 2 1 2 0 6 0 0 0 0
3 1 1 1 1 0 0 0 0 0 0
1 0 1 0 0 0 1 0 0 0 0
1 0 0 1 0 0 1 0 0 0 0
1 0 0 0 1 0 1 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 1 0 0 0
1 0 0 0 0 0 1 0 1 0 0
1 0 1 0 0 0 0 0 0 1 0
1 0 0 0 1 0 0 0 0 0 1
```

in main() after allocate and initialize table.

```
=====
=
ProcUsed:0   currentTime:0
Time:        0       1       2       3       4       5       6       7       8       9       10
-----
-
Proc:1 |    0    |
-----
-
Proc:2 |    0    |
-----
-
Proc:3 |    0    |
-----
-
Open(Time_0):
In scheduling() after loadOpen currentTime is 0
Open(Time_0): (10, 1) --> (9, 1) --> (8, 1) --> (7, 1) --> (5, 1) --> (1, 1)
after loadTable currentTime is 0
=====
=
ProcUsed:3   currentTime:0
Time:        0       1       2       3       4       5       6       7       8       9       10
-----
-
Proc:1 |   10   |
-----
-
Proc:2 |    9   |
-----
-
Proc:3 |    8   |
-----
-
Open(Time_0): (7, 1) --> (5, 1) --> (1, 1)
Open(Time_1): (7, 1) --> (5, 1) --> (1, 1)
In scheduling() after loadOpen currentTime is 1
```

```

Open(Time_1): (7, 1) --> (5, 1) --> (1, 1)
after loadTable currentTime is 1
=====
=
ProcUsed:3   currentTime:1
Time:        0      1      2      3      4      5      6      7      8      9      10
-----
-
Proc:1 | 10 | 7 |
-----
-
Proc:2 | 9 | 5 |
-----
-
Proc:3 | 8 | 1 |
-----
-
Open(Time_1):
Open(Time_2):
In scheduling() after loadOpen currentTime is 2
Open(Time_2): (4, 1) --> (3, 1) --> (2, 1)
after loadTable currentTime is 2
=====
=
ProcUsed:3   currentTime:2
Time:        0      1      2      3      4      5      6      7      8      9      10
-----
-
Proc:1 | 10 | 7 | 4 |
-----
-
Proc:2 | 9 | 5 | 3 |
-----
-
Proc:3 | 8 | 1 | 2 |
-----
-
Open(Time_2):
Open(Time_3):
In scheduling() after loadOpen currentTime is 3
Open(Time_3): (6, 1)
after loadTable currentTime is 3
=====
=
ProcUsed:3   currentTime:3
Time:        0      1      2      3      4      5      6      7      8      9      10
-----
-
Proc:1 | 10 | 7 | 4 | 6 |
-----
-
Proc:2 | 9 | 5 | 3 | 0 |
-----
-
Proc:3 | 8 | 1 | 2 | 0 |
-----
-
Open(Time_3):
Open(Time_4):
In main() printing Table and Open
=====
=
ProcUsed:3   currentTime:4
Time:        0      1      2      3      4      5      6      7      8      9      10
-----
-
Proc:1 | 10 | 7 | 4 | 6 | 0 |
-----
-
Proc:2 | 9 | 5 | 3 | 0 | 0 |
-----

```

-
Proc:3 | 8 | 1 | 2 | 0 | 0 |

-

Open(Time_4):
Open(Time_4):

****deBugFile****

Entering setMatrix method!
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 3
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 2
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 1
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 2
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 6
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 0
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!

```

In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Before leaving setMatrix method, the Matrix below
10 0 2 1 2 0 6 0 0 0 0
3 1 1 1 1 0 0 0 0 0 0
1 0 1 0 0 0 1 0 0 0 0
1 0 0 1 0 0 1 0 0 0 0
1 0 0 0 1 0 1 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 1 0 0 0
1 0 0 0 0 0 1 0 1 0 0
1 0 1 0 0 0 0 0 0 1 0
1 0 0 0 1 0 0 0 0 0 1
Leaving serMatrix().
Entering loadJobTimeAry() method!
1
2
3
4
5
6
7
8
9
10
Leaving loadJobTimeAry() method.
Entering scheduling() method!
Entering loadOpen() method!
Find an orphan, the orphan is 1
Find an orphan, the orphan is 5
Find an orphan, the orphan is 7
Find an orphan, the orphan is 8
Find an orphan, the orphan is 9
Find an orphan, the orphan is 10
Before leaving loadOpen().
Open(Time_0): (10, 1) --> (9, 1) --> (8, 1) --> (7, 1) --> (5, 1) --> (1, 1)
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_0): (10, 1) --> (9, 1) --> (8, 1) --> (7, 1) --> (5, 1) --> (1, 1)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!
in getNextProc() found an available proc#2
Entering getNextProc() method!
in getNextProc() found an available proc#3
Entering getNextProc() method!
Leaving getNextProc() method. No available proc can be found.
Before leaving loadTable().
=====
=====
ProcUsed:3  currentTime:0
Time:      0      1      2      3      4      5      6      7      8      9      10
-----
-----
Proc:1 | 10 |

```


Proc:2 | 9 |

Proc:3 | 8 |

Open(Time_0): (7, 1) --> (5, 1) --> (1, 1)
Open(Time_0): (7, 1) --> (5, 1) --> (1, 1)
Leaving loadTable() method.
Entering deleteDoneJobs() method!
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Leaving deleteDoneJobs() method below is the updated matrix.
7 0 1 1 1 0 5 0 0 0 0
3 2 1 1 1 0 0 0 0 0 0
1 0 1 0 0 0 1 0 0 0 0
1 0 0 1 0 0 1 0 0 0 0
1 0 0 0 1 0 1 0 0 0 0
1 0 0 0 0 2 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 2 0 0 0
1 0 0 0 0 0 1 0 0 0 0
1 0 1 0 0 0 0 0 0 0 0
1 0 0 0 1 0 0 0 0 0 0
Entering loadOpen() method!
Before leaving loadOpen().
Open(Time_1): (7, 1) --> (5, 1) --> (1, 1)
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_1): (7, 1) --> (5, 1) --> (1, 1)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!
in getNextProc() found an available proc#2
Entering getNextProc() method!
in getNextProc() found an available proc#3
Entering getNextProc() method!
Leaving getNextProc() method. No available proc can be found.
Before leaving loadTable().
=====

=====
ProcUsed:3 currentTime:1
Time: 0 1 2 3 4 5 6 7 8 9 10

Proc:1 | 10 | 7 |

Proc:2 | 9 | 5 |

Proc:3 | 8 | 1 |

Open(Time_1):
Open(Time_1):
Leaving loadTable() method.

```

Entering deleteDoneJobs() method!
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Leaving deleteDoneJobs() method below is the updated matrix.
4 0 0 0 0 0 3 0 0 0 0
3 0 1 1 1 0 0 0 0 0 0
1 0 1 0 0 0 1 0 0 0 0
1 0 0 1 0 0 1 0 0 0 0
1 0 0 0 1 0 1 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0
1 0 1 0 0 0 0 0 0 0 0
1 0 0 0 1 0 0 0 0 0 0
Entering loadOpen() method!
Find an orphan, the orphan is 2
Find an orphan, the orphan is 3
Find an orphan, the orphan is 4
Before leaving loadOpen().
Open(Time_2): (4, 1) --> (3, 1) --> (2, 1)
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_2): (4, 1) --> (3, 1) --> (2, 1)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!
in getNextProc() found an available proc#2
Entering getNextProc() method!
in getNextProc() found an available proc#3
Entering getNextProc() method!
Leaving getNextProc() method. No available proc can be found.
Before leaving loadTable().
=====
=====
ProcUsed:3   currentTime:2
Time:        0      1      2      3      4      5      6      7      8      9      10
-----
-----
Proc:1 |    10 |    7 |    4 |
-----
-----
Proc:2 |    9 |    5 |    3 |
-----
-----
Proc:3 |    8 |    1 |    2 |
-----
-----
Open(Time_2):
Open(Time_2):
Leaving loadTable() method.
Entering deleteDoneJobs() method!
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Leaving deleteDoneJobs() method below is the updated matrix.

```

```

1 0 0 0 0 0 0 0 0 0 0
3 0 1 1 1 0 0 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0
1 0 1 0 0 0 0 0 0 0 0
1 0 0 0 1 0 0 0 0 0 0

```

```

Entering loadOpen() method!
Find an orphan, the orphan is 6
Before leaving loadOpen().
Open(Time_3): (6, 1)
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_3): (6, 1)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!
in getNextProc() found an available proc#2
Before leaving loadTable().
=====
=====

```

```

ProcUsed:3   currentTime:3
Time:        0       1       2       3       4       5       6       7       8       9      10

```

Set_2 with unlimited processors

OutFile1

```

In main() after loadMatrix().
0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0

```

```

In main() after setMatrix().
10 0 2 1 2 0 6 0 0 0 0
3 1 1 1 0 0 0 0 0 0 0
1 0 1 0 0 0 1 0 0 0 0
1 0 0 1 0 0 1 0 0 0 0
1 0 0 0 1 0 1 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 1 0 0 0
1 0 0 0 0 0 1 0 1 0 0
1 0 1 0 0 0 0 0 0 1 0
1 0 0 0 1 0 0 0 0 0 1

```

```

In main() after allocate and initialize table.

```

```

ProcUsed:0   currentTime:0
Time:        0       1       2       3       4       5       6       7       8       9      10
12          13          14          15          16          17          18          19          20          21

```

```

Proc:1      | 0       |
-----
Proc:2      | 0       |
-----
Proc:3      | 0       |
-----
Proc:4      | 0       |
-----
Proc:5      | 0       |
-----
Proc:6      | 0       |
-----
Proc:7      | 0       |
-----
Proc:8      | 0       |
-----
Proc:9      | 0       |
-----
Proc:10     | 0       |
-----

```

```

Open(Time_0):
In scheduling() after loadOpen currentTime is 0
Open(Time_0): (1, 1) --> (10, 2) --> (7, 2) --> (9, 3) --> (8, 3) --> (5, 4)
after loadTable currentTime is 0

```

```

ProcUsed:6   currentTime:0
Time:        0       1       2       3       4       5       6       7       8       9      10
12          13          14          15          16          17          18          19          20          21

```

```

Proc:1      | 1       |
-----
Proc:2      | 10      |
-----
Proc:3      | 7       |
-----
Proc:4      | 9       |
-----
Proc:5      | 8       |
-----

```



```
Proc:6      | 5      |
Proc:7      | 0      |
Proc:8      | 0      |
Proc:9      | 0      |
Proc:10     | 0      |
Open(Time_0):
Open(Time_1):
In scheduling() after loadOpen currentTime is 1
Open(Time_1): (3, 2)
after loadTable currentTime is 1
=====
ProcUsed:6  currentTime:1
Time:      0      1      2      3      4      5      6      7      8      9      10
12      13      14      15      16      17      18      19      20      21
Proc:1      | 1      | 3      |
Proc:2      | 10     | 10     |
Proc:3      | 7      | 7      |
Proc:4      | 9      | 9      |
Proc:5      | 8      | 8      |
Proc:6      | 5      | 5      |
Proc:7      | 0      | 0      |
Proc:8      | 0      | 0      |
Proc:9      | 0      | 0      |
Proc:10     | 0      | 0      |
Open(Time_1):
Open(Time_2):
In scheduling() after loadOpen currentTime is 2
Open(Time_2): (4, 1)
after loadTable currentTime is 2
=====
ProcUsed:6  currentTime:2
Time:      0      1      2      3      4      5      6      7      8      9      10
12      13      14      15      16      17      18      19      20      21
Proc:1      | 1      | 3      | 3      |
Proc:2      | 10     | 10     | 4      |
Proc:3      | 7      | 7      | 0      |
Proc:4      | 9      | 9      | 9      |
Proc:5      | 8      | 8      | 8      |
Proc:6      | 5      | 5      | 5      |
Proc:7      | 0      | 0      | 0      |
Proc:8      | 0      | 0      | 0      |
Proc:9      | 0      | 0      | 0      |
Proc:10     | 0      | 0      | 0      |
Open(Time_2):
Open(Time_3):
In scheduling() after loadOpen currentTime is 3
Open(Time_3): (2, 2)
after loadTable currentTime is 3
=====
ProcUsed:6  currentTime:3
Time:      0      1      2      3      4      5      6      7      8      9      10
12      13      14      15      16      17      18      19      20      21
Proc:1      | 1      | 3      | 3      | 2      |
Proc:2      | 10     | 10     | 4      | 0      |
Proc:3      | 7      | 7      | 0      | 0      |
Proc:4      | 9      | 9      | 9      | 0      |
Proc:5      | 8      | 8      | 8      | 0      |
Proc:6      | 5      | 5      | 5      | 5      |
Proc:7      | 0      | 0      | 0      | 0      |
Proc:8      | 0      | 0      | 0      | 0      |
Proc:9      | 0      | 0      | 0      | 0      |
Proc:10     | 0      | 0      | 0      | 0      |
Open(Time_3):
Open(Time_4):
In scheduling() after loadOpen currentTime is 4
Open(Time_4):
after loadTable currentTime is 4
=====
ProcUsed:6  currentTime:4
Time:      0      1      2      3      4      5      6      7      8      9      10
12      13      14      15      16      17      18      19      20      21
Proc:1      | 1      | 3      | 3      | 2      | 2      |
Proc:2      | 10     | 10     | 4      | 0      | 0      |
Proc:3      | 7      | 7      | 0      | 0      | 0      |
Proc:4      | 9      | 9      | 9      | 0      | 0      |
Proc:5      | 8      | 8      | 8      | 0      | 0      |
Proc:6      | 5      | 5      | 5      | 5      | 0      |
Proc:7      | 0      | 0      | 0      | 0      | 0      |
Proc:8      | 0      | 0      | 0      | 0      | 0      |
Proc:9      | 0      | 0      | 0      | 0      | 0      |
Proc:10     | 0      | 0      | 0      | 0      | 0      |
Open(Time_4):
Open(Time_5):
In scheduling() after loadOpen currentTime is 5
Open(Time_5): (6, 1)
after loadTable currentTime is 5
=====
ProcUsed:6  currentTime:5
Time:      0      1      2      3      4      5      6      7      8      9      10
12      13      14      15      16      17      18      19      20      21
Proc:1      | 1      | 3      | 3      | 2      | 2      | 6      |
Proc:2      | 10     | 10     | 4      | 0      | 0      | 0      |
```

```

Proc:3 | 7 | 7 | 0 | 0 | 0 | 0 |
-----
Proc:4 | 9 | 9 | 9 | 0 | 0 | 0 |
-----
Proc:5 | 8 | 8 | 8 | 0 | 0 | 0 |
-----
Proc:6 | 5 | 5 | 5 | 5 | 0 | 0 |
-----
Proc:7 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:8 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:9 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:10 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Open(Time_5):
Open(Time_5):
In main() Printing Table and Open
-----
ProcUsed:6 currentTime:6
Time: 0 1 2 3 4 5 6 7 8 9 10
12 13 14 15 16 17 18 19 20 21
-----
Proc:1 | 1 | 3 | 3 | 2 | 2 | 6 | 0 |
-----
Proc:2 | 10 | 10 | 4 | 0 | 0 | 0 | 0 |
-----
Proc:3 | 7 | 7 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:4 | 9 | 9 | 9 | 0 | 0 | 0 | 0 |
-----
Proc:5 | 8 | 8 | 8 | 0 | 0 | 0 | 0 |
-----
Proc:6 | 5 | 5 | 5 | 5 | 0 | 0 | 0 |
-----
Proc:7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Open(Time_6):
Open(Time_6):

```

****deBugFile****

```

Entering setMatrix method!
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 3
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 2
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 1
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 2
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 6
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 0
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0

```

```

Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Before leaving setMatrix method, the Matrix below
10 0 2 1 2 0 6 0 0 0 0
3 1 1 1 1 0 0 0 0 0 0
1 0 1 0 0 0 1 0 0 0 0
1 0 0 1 0 0 1 0 0 0 0
1 0 0 0 1 0 1 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 1 0 0 0
1 0 0 0 0 0 1 0 1 0 0
1 0 1 0 0 0 0 0 0 1 0
1 0 0 0 1 0 0 0 0 0 1
Leaving serMatrix().
Entering loadJobTimeAry() method!
1
3
5
6
10
11
13
16
19
21
Leaving loadJobTimeAry() method.
Entering scheduling() method!
Entering loadOpen() method!
Find an orphan, the orphan is 1
Find an orphan, the orphan is 5
Find an orphan, the orphan is 7
Find an orphan, the orphan is 8
Find an orphan, the orphan is 9
Find an orphan, the orphan is 10
Before leaving loadOpen().
Open(Time_0): (1, 1) --> (10, 2) --> (7, 2) --> (9, 3) --> (8, 3) --> (5, 4)
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_0): (1, 1) --> (10, 2) --> (7, 2) --> (9, 3) --> (8, 3) --> (5, 4)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!

```

```
in getNextProc() found an available proc#2
Entering getNextProc() method!
in getNextProc() found an available proc#3
Entering getNextProc() method!
in getNextProc() found an available proc#4
Entering getNextProc() method!
in getNextProc() found an available proc#5
Entering getNextProc() method!
in getNextProc() found an available proc#6
Entering getNextProc() method!
in getNextProc() found an available proc#7
Before leaving loadTable().
```

```
=====
=====
```

```
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
```

```
-----
-----
```

```
Proc:1 | 1 |
```

```
-----
-----
```

```
Proc:2 | 10 |
```

```
-----
-----
```

```
Proc:3 | 7 |
```

```
-----
-----
```

```
Proc:4 | 9 |
```

```
-----
-----
```

```
Proc:5 | 8 |
```

```
-----
-----
```

```
Proc:6 | 5 |
```

```
-----
-----
```

```
Proc:7 | 0 |
```

```
-----
-----
```

```
Proc:8 | 0 |
```

```
-----
-----
```

```
Proc:9 | 0 |
```

```
-----
-----
```

```
Proc:10      | 0 |
```

```
-----
-----
```

```
Open(Time_0):
```

```
Open(Time_0):
```

```

Leaving loadTable() method.
Entering deleteDoneJobs() method!
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Leaving deleteDoneJobs() method below is the updated matrix.
9 0 1 0 1 0 6 0 0 0 0
3 0 1 1 1 0 0 0 0 0 0
1 0 1 0 0 0 1 0 0 0 0
1 0 0 1 0 0 1 0 0 0 0
1 0 0 0 1 0 1 0 0 0 0
1 0 0 0 0 2 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0
1 0 0 0 0 0 1 2 0 0 0
1 0 0 0 0 0 1 0 2 0 0
1 0 1 0 0 0 0 0 0 2 0
1 0 0 0 1 0 0 0 0 0 2
Entering loadOpen() method!
Find an orphan, the orphan is 3
Before leaving loadOpen().
Open(Time_1): (3, 2)
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_1): (3, 2)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!
in getNextProc() found an available proc#7
Before leaving loadTable().
=====
=====
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
-----
-----
-----
Proc:1 |    1    |    3    |
-----
-----
-----
Proc:2 |   10    |   10    |
-----
-----
-----
Proc:3 |    7    |    7    |
-----
-----
-----
Proc:4 |    9    |    9    |
-----
-----
-----
Proc:5 |    8    |    8    |
-----
-----
-----
Proc:6 |    5    |    5    |
-----
-----
***Set_3 with 3 processors***
**OutFile1**
In main() after loadMatrix().

```

```

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0

```

```

In main() after setMatrix().
14 0 1 1 1 1 5 5 4 1 0 0 0 0 2
3 1 1 1 1 0 0 0 0 0 0 0 0 0 0
2 0 1 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 1 0 0 1 1 0 0 0 0 0 0 0
1 0 0 0 1 0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0 0 0 0 0
2 0 0 0 0 0 1 0 1 0 0 0 0 0 1
1 0 0 0 0 0 0 1 0 1 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
2 0 0 0 0 0 1 0 0 0 1 0 0 0 1
2 0 0 0 0 0 0 1 1 0 0 1 0 0 0
1 0 0 0 0 0 0 0 1 0 0 0 1 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 1 0
1 0 0 0 0 1 0 0 0 0 0 0 0 0 1

```

in main() after allocate and initialize table.

```

=====
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
=====

```

```

Proc:1  |  0  |
=====

```

```

Proc:2  |  0  |
=====

```

```

Proc:3  |  0  |
=====

```

```

Open(Time_0):
In scheduling() after loadOpen currentTime is 0
Open(Time_0): (12, 1) --> (10, 1) --> (11, 2) --> (1, 2) --> (13, 4)
after loadTable currentTime is 0
=====

```

```

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  |  12  |
=====

```

```

Proc:2  |  10  |
=====

```

```

Proc:3  |  11  |
=====

```

```

Open(Time_0): (1, 2) --> (13, 4)
Open(Time_1): (1, 2) --> (13, 4)
In scheduling() after loadOpen currentTime is 1
Open(Time_1): (1, 2) --> (13, 4)
after loadTable currentTime is 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      27
=====

```

```

Proc:1  |  12  |  1  |
=====

```

```

Proc:2  |  10  |  13  |
=====

```

```

Proc:3  |  11  |  11  |
=====

```

Open(Time_1):

Open(Time_2):
In scheduling() after loadOpen currentTime is 2
Open(Time_2):
after loadTable currentTime is 2

```
=====
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  | 12 | 1 | 1 |
-----
```

```
-----
Proc:2  | 10 | 13 | 13 |
-----
```

```
-----
Proc:3  | 11 | 11 | 0 |
-----
```

Open(Time_2):
Open(Time_3):
In scheduling() after loadOpen currentTime is 3
Open(Time_3): (3, 1) --> (4, 2) --> (2, 3)
after loadTable currentTime is 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     27
=====
```

```
-----
Proc:1  | 12 | 1 | 1 | 3 |
-----
```

```
-----
Proc:2  | 10 | 13 | 13 | 13 |
-----
```

```
-----
Proc:3  | 11 | 11 | 0 | 4 |
-----
```

Open(Time_3): (2, 3)
Open(Time_4): (2, 3)
In scheduling() after loadOpen currentTime is 4
Open(Time_4): (2, 3)
after loadTable currentTime is 4

```
=====
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  | 12 | 1 | 1 | 3 | 2 |
-----
```

```
-----
Proc:2  | 10 | 13 | 13 | 13 | 13 |
-----
```

```
-----
Proc:3  | 11 | 11 | 0 | 4 | 4 |
-----
```

Open(Time_4):
Open(Time_5):
In scheduling() after loadOpen currentTime is 5
Open(Time_5):
after loadTable currentTime is 5

```
=====
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  | 12 | 1 | 1 | 3 | 2 | 2 |
-----
```

```
-----
Proc:2  | 10 | 13 | 13 | 13 | 13 | 0 |
-----
```

```
-----
Proc:3  | 11 | 11 | 0 | 4 | 4 | 0 |
-----
```

Open(Time_5):
Open(Time_6):
In scheduling() after loadOpen currentTime is 6
Open(Time_6):
after loadTable currentTime is 6

```
=====
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 | 12 | 1 | 1 | 3 | 2 | 2 | 2 |
-----
Proc:2 | 10 | 13 | 13 | 13 | 13 | 0 | 0 |
-----
Proc:3 | 11 | 11 | 0 | 4 | 4 | 0 | 0 |
-----

Open(Time_6):
Open(Time_7):
In scheduling() after loadOpen currentTime is 7
Open(Time_7):
after loadTable currentTime is 7
=====
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 | 12 | 1 | 1 | 3 | 2 | 2 | 2 | 0 |
-----
Proc:2 | 10 | 13 | 13 | 13 | 13 | 0 | 0 | 0 |
-----
Proc:3 | 11 | 11 | 0 | 4 | 4 | 0 | 0 | 0 |
-----

Open(Time_7):
Open(Time_8):

```

*****There is cycle in the graph!!!*****

```

In main() printing Table and Open
=====
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 | 12 | 1 | 1 | 3 | 2 | 2 | 2 | 0 | 0 |
-----
Proc:2 | 10 | 13 | 13 | 13 | 13 | 0 | 0 | 0 | 0 |
-----
Proc:3 | 11 | 11 | 0 | 4 | 4 | 0 | 0 | 0 | 0 |
-----

Open(Time_8):
Open(Time_8):

```

****debugFile****

```

Entering setMatrix method!
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 3
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 1
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 2
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 1
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 2

```


[illegible]

```

Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Before leaving setMatrix method, the Matrix below
14 0 1 1 1 1 5 5 4 1 0 0 0 0 2
3 1 1 1 1 0 0 0 0 0 0 0 0 0 0
2 0 1 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 1 0 0 1 1 0 0 0 0 0 0 0
1 0 0 0 1 0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0 0 0 0 0
2 0 0 0 0 0 1 0 1 0 0 0 0 0 1
1 0 0 0 0 0 0 1 0 1 0 0 0 0 0
1 0 0 0 0 0 0 1 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
2 0 0 0 0 0 1 0 0 0 1 0 0 0 1
2 0 0 0 0 0 0 1 1 0 0 1 0 0 0
1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 1 0
1 0 0 0 0 1 0 0 0 0 0 0 0 0 1
Leaving serMatrix().
Entering loadJobTimeAry() method!
2
5
6
8
9
11
12
14
17
18
20
21
25
27
Leaving loadJobTimeAry() method.
Entering scheduling() method!
Entering loadOpen() method!
Find an orphan, the orphan is 1
Find an orphan, the orphan is 10
Find an orphan, the orphan is 11
Find an orphan, the orphan is 12
Find an orphan, the orphan is 13
Before leaving loadOpen().
Open(Time_0): (12, 1) --> (10, 1) --> (11, 2) --> (1, 2) --> (13, 4)
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_0): (12, 1) --> (10, 1) --> (11, 2) --> (1, 2) --> (13, 4)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!
in getNextProc() found an available proc#2
Entering getNextProc() method!
in getNextProc() found an available proc#3
Entering getNextProc() method!
Leaving getNextProc() method. No available proc can be found.
Before leaving loadTable().
=====
=====
=====
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 | 12 |
-----
-----
-----
```

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

```
-----
Proc:3 | 11 |
-----
-----
-----
```

```
-----
Open(Time_0): (1, 2) --> (13, 4)
Open(Time_0): (1, 2) --> (13, 4)
Leaving loadTable() method.
Entering deleteDoneJobs() method!
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Leaving deleteDoneJobs() method below is the updated matrix.
12 0 1 1 1 1 4 5 3 1 0 0 0 0 1
3 2 1 1 1 0 0 0 0 0 0 0 0 0 0
2 0 1 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 1 0 0 1 1 0 0 0 0 0 0 0
1 0 0 0 1 0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0 0 0 0 0
2 0 0 0 0 0 1 0 1 0 0 0 0 0 1
1 0 0 0 0 0 0 1 0 1 0 0 0 0 0
1 0 0 0 0 0 0 1 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
2 0 0 0 0 0 0 1 0 0 0 0 0 0 1
2 0 0 0 0 0 0 1 1 0 0 2 0 0 0
1 0 0 0 0 0 0 0 1 0 0 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 2 0
1 0 0 0 0 1 0 0 0 0 0 0 0 0 1
Entering loadOpen() method!
Before leaving loadOpen().
Open(Time_1): (1, 2) --> (13, 4)
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_1): (1, 2) --> (13, 4)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!
in getNextProc() found an available proc#2
Entering getNextProc() method!
Leaving getNextProc() method. No available proc can be found.
Before leaving loadTable().
=====
=====
=====
```

```
ProcUsed:3  currentTime:1
```

*****Set_4 with 3 processors*****

****OutFile1****

In main() after loadMatrix().

```

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

```

In main() after setMatrix().
14 0 1 1 1 0 5 4 4 1 0 0 0 0 2
3 1 1 1 1 0 0 0 0 0 0 0 0 0 0
2 0 1 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 1 0 0 1 1 0 0 0 0 0 0 0
1 0 0 0 1 0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0 0 0 0 0
2 0 0 0 0 0 1 0 1 0 0 0 0 0 1
1 0 0 0 0 0 0 1 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
2 0 0 0 0 0 1 0 0 0 1 0 0 0 1
2 0 0 0 0 0 0 1 1 0 0 1 0 0 0
1 0 0 0 0 0 0 0 1 0 0 0 1 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

```

in main() after allocate and initialize table.

```

=====
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      28
=====

```

```

Proc:1  |  0  |
=====

```

```

Proc:2  |  0  |
=====

```

```

Proc:3  |  0  |
=====

```

```

Open(Time_0):
In scheduling() after loadOpen currentTime is 0
Open(Time_0): (13, 1) --> (10, 1) --> (5, 1) --> (12, 2) --> (1, 2) --> (11, 4)
after loadTable currentTime is 0
=====

```

```

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      28
=====

```

```

Proc:1  |  13  |
=====

```

```

Proc:2  |  10  |
=====

```

```

Proc:3  |  5  |
=====

```

```

Open(Time_0): (12, 2) --> (1, 2) --> (11, 4)
Open(Time_1): (12, 2) --> (1, 2) --> (11, 4)
In scheduling() after loadOpen currentTime is 1
Open(Time_1): (12, 2) --> (1, 2) --> (11, 4)
after loadTable currentTime is 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      27      28
=====

```

```

Proc:1  |  13  |  12  |
=====

```

```

Proc:2  |  10  |  1  |
=====

```

```

Proc:3  |  5  |  11  |
=====

```

Open(Time_1):

```

Open(Time_2):
In scheduling() after loadOpen currentTime is 2
Open(Time_2):
after loadTable currentTime is 2

```

```

=====
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     28

```

```

-----
Proc:1  | 13  | 12  | 12  |
-----

```

```

-----
Proc:2  | 10  | 1   | 1   |
-----

```

```

-----
Proc:3  | 5   | 11  | 11  |
-----

```

```

Open(Time_2):
Open(Time_3):
In scheduling() after loadOpen currentTime is 3
Open(Time_3): (3, 1) --> (4, 2) --> (2, 3)
after loadTable currentTime is 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     27     28

```

```

-----
Proc:1  | 13  | 12  | 12  | 3   |
-----

```

```

-----
Proc:2  | 10  | 1   | 1   | 4   |
-----

```

```

-----
Proc:3  | 5   | 11  | 11  | 11  |
-----

```

```

Open(Time_3): (2, 3)
Open(Time_4): (2, 3)
In scheduling() after loadOpen currentTime is 4
Open(Time_4): (2, 3)
after loadTable currentTime is 4

```

```

=====
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     28

```

```

-----
Proc:1  | 13  | 12  | 12  | 3   | 2   |
-----

```

```

-----
Proc:2  | 10  | 1   | 1   | 4   | 4   |
-----

```

```

-----
Proc:3  | 5   | 11  | 11  | 11  | 11  |
-----

```

```

Open(Time_4):
Open(Time_5):
In scheduling() after loadOpen currentTime is 5
Open(Time_5):
after loadTable currentTime is 5

```

```

=====
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     28

```

```

-----
Proc:1  | 13  | 12  | 12  | 3   | 2   | 2   |
-----

```

```

-----
Proc:2  | 10  | 1   | 1   | 4   | 4   | 0   |
-----

```

```

-----
Proc:3  | 5   | 11  | 11  | 11  | 11  | 0   |
-----

```

```

Open(Time_5):
Open(Time_6):
In scheduling() after loadOpen currentTime is 6
Open(Time_6):
after loadTable currentTime is 6

```

```

=====
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	28									

Proc:1		13		12		12		3		2		2		2	
--------	--	----	--	----	--	----	--	---	--	---	--	---	--	---	--

Proc:2		10		1		1		4		4		0		0	
--------	--	----	--	---	--	---	--	---	--	---	--	---	--	---	--

Proc:3		5		11		11		11		11		0		0	
--------	--	---	--	----	--	----	--	----	--	----	--	---	--	---	--

Open(Time_6):
Open(Time_7):
In scheduling() after loadOpen currentTime is 7
Open(Time_7): (7, 1) --> (6, 3)
after loadTable currentTime is 7

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	28									

Proc:1		13		12		12		3		2		2		2		7	
--------	--	----	--	----	--	----	--	---	--	---	--	---	--	---	--	---	--

Proc:2		10		1		1		4		4		0		0		6	
--------	--	----	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--

Proc:3		5		11		11		11		11		0		0		0	
--------	--	---	--	----	--	----	--	----	--	----	--	---	--	---	--	---	--

Open(Time_7):
Open(Time_8):
In scheduling() after loadOpen currentTime is 8
Open(Time_8): (9, 3)
after loadTable currentTime is 8

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	28									

Proc:1		13		12		12		3		2		2		2		7		9	
--------	--	----	--	----	--	----	--	---	--	---	--	---	--	---	--	---	--	---	--

Proc:2		10		1		1		4		4		0		0		6		6	
--------	--	----	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--

Proc:3		5		11		11		11		11		0		0		0		0	
--------	--	---	--	----	--	----	--	----	--	----	--	---	--	---	--	---	--	---	--

Open(Time_8):
Open(Time_9):
In scheduling() after loadOpen currentTime is 9
Open(Time_9):
after loadTable currentTime is 9

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	28									

Proc:1		13		12		12		3		2		2		2		7		9		9	
--------	--	----	--	----	--	----	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--

Proc:2		10		1		1		4		4		0		0		6		6		6	
--------	--	----	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--

Proc:3		5		11		11		11		11		0		0		0		0		0	
--------	--	---	--	----	--	----	--	----	--	----	--	---	--	---	--	---	--	---	--	---	--

Open(Time_9):
Open(Time_10):
In scheduling() after loadOpen currentTime is 10
Open(Time_10): (14, 2)
after loadTable currentTime is 10

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	28									

Proc:1		13		12		12		3		2		2		2		7		9		9		9	
--------	--	----	--	----	--	----	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--

```

-----
Proc:2 | 10 | 1 | 1 | 4 | 4 | 0 | 0 | 6 | 6 | 6 | 14 |
-----
Proc:3 | 5 | 11 | 11 | 11 | 11 | 0 | 0 | 0 | 0 | 0 | 0 |
-----

Open(Time_10):
Open(Time_11):
In scheduling() after loadOpen currentTime is 11
Open(Time_11): (8, 2)
after loadTable currentTime is 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     28
-----
Proc:1 | 13 | 12 | 12 | 3 | 2 | 2 | 2 | 7 | 9 | 9 | 9 | 8 |
|
-----
Proc:2 | 10 | 1 | 1 | 4 | 4 | 0 | 0 | 6 | 6 | 6 | 14 | 14 |
|
-----
Proc:3 | 5 | 11 | 11 | 11 | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|
-----

Open(Time_11):
Open(Time_12):
In scheduling() after loadOpen currentTime is 12
Open(Time_12):
after loadTable currentTime is 12
=====
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     28
-----
Proc:1 | 13 | 12 | 12 | 3 | 2 | 2 | 2 | 7 | 9 | 9 | 9 | 8 |
| 8 |
-----
Proc:2 | 10 | 1 | 1 | 4 | 4 | 0 | 0 | 6 | 6 | 6 | 14 | 14 |
| 0 |
-----
Proc:3 | 5 | 11 | 11 | 11 | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 |
-----

Open(Time_12):
Open(Time_13):
In main() printing Table and Open
=====
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     28
-----
Proc:1 | 13 | 12 | 12 | 3 | 2 | 2 | 2 | 7 | 9 | 9 | 9 | 8 |
| 8 | 0 |
-----
Proc:2 | 10 | 1 | 1 | 4 | 4 | 0 | 0 | 6 | 6 | 6 | 14 | 14 |
| 0 | 0 |
-----
Proc:3 | 5 | 11 | 11 | 11 | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 |
-----

Open(Time_13):
Open(Time_13):

```

****deBugFile****

```

Entering setMatrix method!
Entering countParent() method!
In countParent() parent count is 0

```

[illegible]


```

Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 2
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 0
Leaving countDependent() method.
Before leaving setMatrix method, the Matrix below
14 0 1 1 1 0 5 4 4 1 0 0 0 0 2
3 1 1 1 1 0 0 0 0 0 0 0 0 0 0
2 0 1 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 1 0 0 1 1 0 0 0 0 0 0 0
1 0 0 0 1 0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0 0 0 0 0
2 0 0 0 0 0 1 0 1 0 0 0 0 0 1
1 0 0 0 0 0 0 1 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
2 0 0 0 0 0 1 0 0 0 1 0 0 0 1
2 0 0 0 0 0 0 1 1 0 0 1 0 0 0
1 0 0 0 0 0 0 0 1 0 0 0 1 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
Leaving serMatrix().
Entering loadJobTimeAry() method!
2
5
6
8
9
12
13
15
18
19
23
25
26
28
Leaving loadJobTimeAry() method.
Entering scheduling() method!
Entering loadOpen() method!
Find an orphan, the orphan is 1
Find an orphan, the orphan is 5
Find an orphan, the orphan is 10
Find an orphan, the orphan is 11
Find an orphan, the orphan is 12
Find an orphan, the orphan is 13
Before leaving loadOpen().
Open(Time_0): (13, 1) --> (10, 1) --> (5, 1) --> (12, 2) --> (1, 2) --> (11, 4)
Leaving loadOpen() method.

```

```

Entering loadTable() method!
Open(Time_0): (13, 1) --> (10, 1) --> (5, 1) --> (12, 2) --> (1, 2) --> (11, 4)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!
in getNextProc() found an available proc#2
Entering getNextProc() method!
in getNextProc() found an available proc#3
Entering getNextProc() method!
Leaving getNextProc() method. No available proc can be found.
Before leaving loadTable().
=====
=====
=====
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     28
-----
-----
-----
Proc:1 |    13 |
-----
-----
-----
Proc:2 |    10 |
-----
-----
-----
Proc:3 |     5 |
-----
-----
-----
Open(Time_0): (12, 2) --> (1, 2) --> (11, 4)
Open(Time_0): (12, 2) --> (1, 2) --> (11, 4)
Leaving loadTable() method.
Entering deleteDoneJobs() method!
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Leaving deleteDoneJobs() method below is the updated matrix.
11 0 1 1 1 0 2 4 4 1 0 0 0 0 1
3 2 1 1 1 0 0 0 0 0 0 0 0 0 0
2 0 1 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 1 0 0 1 1 0 0 0 0 0 0 0
1 0 0 0 1 0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 0 0
2 0 0 0 0 0 1 0 1 0 0 0 0 0 1
1 0 0 0 0 0 0 1 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
2 0 0 0 0 0 1 0 0 0 0 0 0 0 1
2 0 0 0 0 0 0 1 1 0 0 2 0 0 0
1 0 0 0 0 0 0 0 1 0 0 0 2 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
Entering loadOpen() method!
Before leaving loadOpen().
Open(Time_1): (12, 2) --> (1, 2) --> (11, 4)
Leaving loadOpen() method.

```

```

Entering loadTable() method!
Open(Time_1): (12, 2) --> (1, 2) --> (11, 4)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!
in getNextProc() found an available proc#2
Entering getNextProc() method!
in getNextProc() found an available proc#3
Entering getNextProc() method!
Leaving getNextProc() method. No available proc can be found.
Before leaving loadTable().
=====
=====
=====
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     28
-----
-----
Proc:1 |    13 |    12 |
-----
-----
Proc:2 |    10 |     1 |
-----
-----
Proc:3 |     5 |    11 |
-----
-----
Open(Time_1):
Open(Time_1):
Leaving loadTable() method.
Entering deleteDoneJobs() method!
Leaving deleteDoneJobs() method below is the updated matrix.
11 0 1 1 1 0 2 4 4 1 0 0 0 0 1
3 2 1 1 1 0 0 0 0 0 0 0 0 0 0
2 0 1 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 1 0 0 1 1 0 0 0 0 0 0 0
1 0 0 0 1 0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 0 0
2 0 0 0 0 0 1 0 1 0 0 0 0 0 1
1 0 0 0 0 0 1 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
2 0 0 0 0 0 1 0 0 0 0 0 0 0 1
2 0 0 0 0 0 1 1 0 0 2 0 0 0 0
1 0 0 0 0 0 0 0 1 0 0 0 2 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
Entering loadOpen() method!
Before leaving loadOpen().
Open(Time_2):
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_2):
***Set 4 with unlimited processors***
**OutFile1**
In main() after loadMatrix().

```

```

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

```

In main() after setMatrix().
14 0 1 1 1 0 5 4 4 1 0 0 0 0 2
3 1 1 1 1 0 0 0 0 0 0 0 0 0 0
2 0 1 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 1 0 0 1 1 0 0 0 0 0 0 0
1 0 0 0 1 0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0 0 0 0 0
2 0 0 0 0 0 1 0 1 0 0 0 0 0 1
1 0 0 0 0 0 0 1 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
2 0 0 0 0 0 1 0 0 0 1 0 0 0 1
2 0 0 0 0 0 0 1 1 0 0 1 0 0 0
1 0 0 0 0 0 0 0 1 0 0 0 1 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

```

in main() after allocate and initialize table.

```

=====
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        28
=====

```

```

Proc:1  |  0  |
=====

```

```

Proc:2  |  0  |
=====

```

```

Proc:3  |  0  |
=====

```

```

Proc:4  |  0  |
=====

```

```

Proc:5  |  0  |
=====

```

```

Proc:6  |  0  |
=====

```

```

Proc:7  |  0  |
=====

```

```

Proc:8  |  0  |
=====

```

```

Proc:9  |  0  |
=====

```

```

Proc:10 |  0  |
=====

```

```

Proc:11 |  0  |
=====

```

```

Proc:12 |  0  |
=====

```

```

Proc:13 |  0  |
=====

```

```

Proc:14 |  0  |
=====

```

```

Open(Time_0):
In scheduling() after loadOpen currentTime is 0
Open(Time_0): (13, 1) --> (10, 1) --> (5, 1) --> (12, 2) --> (1, 2) --> (11, 4)
after loadTable currentTime is 0
=====

```

```

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        28

```

Proc:1 | 13 |

Proc:2 | 10 |

Proc:3 | 5 |

Proc:4 | 12 |

Proc:5 | 1 |

Proc:6 | 11 |

Proc:7 | 0 |

Proc:8 | 0 |

Proc:9 | 0 |

Proc:10 | 0 |

Proc:11 | 0 |

Proc:12 | 0 |

Proc:13 | 0 |

Proc:14 | 0 |

Open(Time_0):
Open(Time_1):
In scheduling() after loadOpen currentTime is 1
Open(Time_1):
after loadTable currentTime is 1
=====

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 28

Proc:1 | 13 | 0 |

Proc:2 | 10 | 0 |

Proc:3 | 5 | 0 |

Proc:4 | 12 | 12 |

Proc:5 | 1 | 1 |

Proc:6 | 11 | 11 |

Proc:7 | 0 | 0 |

Proc:8 | 0 | 0 |

Proc:9 | 0 | 0 |

Proc:10 | 0 | 0 |

Proc:11 | 0 | 0 |

Proc:12 | 0 | 0 |

Proc:13 | 0 | 0 |

```
-----
Proc:14 | 0 | 0 |
-----
```

```
-----
Open(Time_1):
Open(Time_2):
In scheduling() after loadOpen currentTime is 2
Open(Time_2): (3, 1) --> (4, 2) --> (2, 3)
after loadTable currentTime is 2
=====
```

```
-----
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     28
-----
```

```
-----
Proc:1 | 13 | 0 | 3 |
-----
```

```
-----
Proc:2 | 10 | 0 | 4 |
-----
```

```
-----
Proc:3 | 5 | 0 | 2 |
-----
```

```
-----
Proc:4 | 12 | 12 | 0 |
-----
```

```
-----
Proc:5 | 1 | 1 | 0 |
-----
```

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

```
-----
Proc:7 | 0 | 0 | 0 |
-----
```

```
-----
Proc:8 | 0 | 0 | 0 |
-----
```

```
-----
Proc:9 | 0 | 0 | 0 |
-----
```

```
-----
Proc:10 | 0 | 0 | 0 |
-----
```

```
-----
Proc:11 | 0 | 0 | 0 |
-----
```

```
-----
Proc:12 | 0 | 0 | 0 |
-----
```

```
-----
Proc:13 | 0 | 0 | 0 |
-----
```

```
-----
Proc:14 | 0 | 0 | 0 |
-----
```

```
-----
Open(Time_2):
Open(Time_3):
In scheduling() after loadOpen currentTime is 3
Open(Time_3):
after loadTable currentTime is 3
=====
```

```
-----
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     28
-----
```

```
-----
Proc:1 | 13 | 0 | 3 | 0 |
-----
```

```
-----
Proc:2 | 10 | 0 | 4 | 4 |
-----
```

```
-----
Proc:3 | 5 | 0 | 2 | 2 |
-----
```

```
-----
Proc:4 | 12 | 12 | 0 | 0 |
-----
```

```
-----
Proc:5 | 1 | 1 | 0 | 0 |
-----
```

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

```
-----
Proc:7 | 0 | 0 | 0 | 0 |
-----
```

```
-----
Proc:8 | 0 | 0 | 0 | 0 |
-----
```

```

-----
Proc:9   | 0   | 0   | 0   | 0   |
-----
Proc:10  | 0   | 0   | 0   | 0   |
-----
Proc:11  | 0   | 0   | 0   | 0   |
-----
Proc:12  | 0   | 0   | 0   | 0   |
-----
Proc:13  | 0   | 0   | 0   | 0   |
-----
Proc:14  | 0   | 0   | 0   | 0   |
-----

```

```

Open(Time_3):
Open(Time_4):
In scheduling() after loadOpen currentTime is 4
Open(Time_4):
after loadTable currentTime is 4
=====

```

```

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  28

```

```

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

```

```

Open(Time_4):
Open(Time_5):
In scheduling() after loadOpen currentTime is 5
Open(Time_5): (7, 1) --> (6, 3)
after loadTable currentTime is 5
=====

```

```

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  28

```

```

-----
Proc:1   | 13  | 0   | 3   | 0   | 0   | 7   |
-----
Proc:2   | 10  | 0   | 4   | 4   | 0   | 6   |
-----
Proc:3   | 5   | 0   | 2   | 2   | 2   | 0   |
-----

```

```

-----
Proc:4 | 12 | 12 | 0 | 0 | 0 | 0 |
-----
Proc:5 | 1 | 1 | 0 | 0 | 0 | 0 |
-----
Proc:6 | 11 | 11 | 11 | 11 | 0 | 0 |
-----
Proc:7 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:8 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:9 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:10 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:11 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:12 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:13 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:14 | 0 | 0 | 0 | 0 | 0 | 0 |
-----

```

```

Open(Time_5):
Open(Time_6):
In scheduling() after loadOpen currentTime is 6
Open(Time_6): (9, 3)
after loadTable currentTime is 6
=====

```

```

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     28

```

```

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

```

```

Open(Time_6):
Open(Time_7):
In scheduling() after loadOpen currentTime is 7
Open(Time_7):
after loadTable currentTime is 7

```



```

=====
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     28
-----
Proc:1  | 13 | 0 | 3 | 0 | 0 | 7 | 9 | 9 |
-----
Proc:2  | 10 | 0 | 4 | 4 | 0 | 6 | 6 | 6 |
-----
Proc:3  | 5 | 0 | 2 | 2 | 2 | 0 | 0 | 0 |
-----
Proc:4  | 12 | 12 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:5  | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:6  | 11 | 11 | 11 | 11 | 0 | 0 | 0 | 0 |
-----
Proc:7  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:8  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:9  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Open(Time_7):
Open(Time_8):
In scheduling() after loadOpen currentTime is 8
Open(Time_8): (14, 2)
after loadTable currentTime is 8
=====

```

```

=====
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     28
-----
Proc:1  | 13 | 0 | 3 | 0 | 0 | 7 | 9 | 9 | 9 |
-----
Proc:2  | 10 | 0 | 4 | 4 | 0 | 6 | 6 | 6 | 14 |
-----
Proc:3  | 5 | 0 | 2 | 2 | 2 | 0 | 0 | 0 | 0 |
-----
Proc:4  | 12 | 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:5  | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:6  | 11 | 11 | 11 | 11 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:7  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:8  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:9  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Proc:11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----

```

Proc:12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Open(Time_8):
Open(Time_9):
In scheduling() after loadOpen currentTime is 9
Open(Time_9): (8, 2)
after loadTable currentTime is 9
=====

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 28

Proc:1 | 13 | 0 | 3 | 0 | 0 | 7 | 9 | 9 | 9 | 8 |

Proc:2 | 10 | 0 | 4 | 4 | 0 | 6 | 6 | 6 | 14 | 14 |

Proc:3 | 5 | 0 | 2 | 2 | 2 | 0 | 0 | 0 | 0 | 0 |

Proc:4 | 12 | 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:5 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:6 | 11 | 11 | 11 | 11 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Open(Time_9):
Open(Time_10):
In scheduling() after loadOpen currentTime is 10
Open(Time_10):
after loadTable currentTime is 10
=====

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 28

Proc:1 | 13 | 0 | 3 | 0 | 0 | 7 | 9 | 9 | 9 | 8 | 8 |

Proc:2 | 10 | 0 | 4 | 4 | 0 | 6 | 6 | 6 | 14 | 14 | 0 |

Proc:3 | 5 | 0 | 2 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:4 | 12 | 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:5 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Proc:6 | 11 | 11 | 11 | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

[illegible]

[illegible]

```

Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 2
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 0
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 1
Leaving countDependent() method.
Entering countParent() method!
In countParent() parent count is 2
Leaving countParent() method.
Entering countDependent() method!
In countDependent() dependent count is 0
Leaving countDependent() method.
Before leaving setMatrix method, the Matrix below
14 0 1 1 1 0 5 4 4 1 0 0 0 0 2
3 1 1 1 1 0 0 0 0 0 0 0 0 0 0
2 0 1 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 1 0 0 1 1 0 0 0 0 0 0 0
1 0 0 0 1 0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 1 1 0 0 0 0 0 0 0 0
2 0 0 0 0 0 1 0 1 0 0 0 0 0 1
1 0 0 0 0 0 0 1 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
2 0 0 0 0 0 1 0 0 0 1 0 0 0 1
2 0 0 0 0 0 0 1 1 0 0 1 0 0 0
1 0 0 0 0 0 0 0 1 0 0 0 1 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
Leaving serMatrix().
Entering loadJobTimeAry() method!
2
5
6
8
9
12
13
15
18
19
23
25
26
28
Leaving loadJobTimeAry() method.
Entering scheduling() method!
Entering loadOpen() method!
Find an orphan, the orphan is 1
Find an orphan, the orphan is 5
Find an orphan, the orphan is 10

```

```

Find an orphan, the orphan is 11
Find an orphan, the orphan is 12
Find an orphan, the orphan is 13
Before leaving loadOpen().
Open(Time_0): (13, 1) --> (10, 1) --> (5, 1) --> (12, 2) --> (1, 2) --> (11, 4)
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_0): (13, 1) --> (10, 1) --> (5, 1) --> (12, 2) --> (1, 2) --> (11, 4)
Entering getNextProc() method!
in getNextProc() found an available proc#1
Entering getNextProc() method!
in getNextProc() found an available proc#2
Entering getNextProc() method!
in getNextProc() found an available proc#3
Entering getNextProc() method!
in getNextProc() found an available proc#4
Entering getNextProc() method!
in getNextProc() found an available proc#5
Entering getNextProc() method!
in getNextProc() found an available proc#6
Entering getNextProc() method!
in getNextProc() found an available proc#7
Before leaving loadTable().

```

```

=====
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   28

```

```

-----
Proc:1 |    13 |
-----

```

```

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

```

```

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

```

```

-----
Proc:4 |    12 |
-----

```

```

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

```

```

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

```

```

-----
Proc:7 |     0 |
-----

```

```

-----
Proc:8 |     0 |

```

```
-----
-----
Proc:9 | 0 |
-----
-----
```

```
-----
Proc:10 | 0 |
-----
-----
```

```
-----
Proc:11 | 0 |
-----
-----
```

```
-----
Proc:12 | 0 |
-----
-----
```

```
-----
Proc:13 | 0 |
-----
-----
```

```
-----
Proc:14 | 0 |
-----
-----
```

```
-----
Open(Time_0):
Open(Time_0):
Leaving loadTable() method.
Entering deleteDoneJobs() method!
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Entering deleteOneJob() method!
Leaving deleteOneJob() method.
Leaving deleteDoneJobs() method below is the updated matrix.
11 0 1 1 1 0 2 4 4 1 0 0 0 0 1
3 2 1 1 1 0 0 0 0 0 0 0 0 0 0
2 0 1 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 1 0 0 1 1 0 0 0 0 0 0 0
1 0 0 0 1 0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 0 0
2 0 0 0 0 0 1 0 1 0 0 0 0 0 1
1 0 0 0 0 0 0 1 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
2 0 0 0 0 0 1 0 0 0 0 0 0 0 1
2 0 0 0 0 0 0 1 1 0 0 2 0 0 0
1 0 0 0 0 0 0 0 1 0 0 0 2 0 0
1 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
Entering loadOpen() method!
Before leaving loadOpen().
Open(Time_1):
Leaving loadOpen() method.
Entering loadTable() method!
Open(Time_1):
```