**EXPERIMENT NO. 04**

|  |
| --- |
| NAME: Ayush Vinod Upadhyay  ROLL NO: I025  SAP ID: 60003220131  BRANCH: Information Technology  BATCH: 1 |

import java.util.\*;

class Exp4Q1.java

{

public

    static void main(String[] args)

    {

        int[][] allocation = {{1, 0, 1}, {2, 1, 2}, {3, 0, 0}, {1, 0, 1}};

        int[][] max = {{2, 1, 1}, {5, 4, 4}, {3, 1, 1}, {1, 1, 1}};

        int[] available = {2, 1, 1};

        int[][] need = new int[4][3];

        for (int i = 0; i < allocation.length; i++)

        {

            for (int j = 0; j < available.length; j++)

            {

                need[i][j] = max[i][j] - allocation[i][j];

            }

        }

        int[] work = available;

        boolean[] finish = new boolean[max.length];

        for (int i = 0; i < work.length; i++)

        {

            finish[i] = false;

        }

        int h = 0;

        int t = 0;

        System.out.println("Sequence");

        while (h <= 4)

        {

            for (int i = 0; i < max.length; i++)

            {

                if (finish[i] == false)

                {

                    if (need[i][0] <= work[0] && need[i][1] <= work[1] && need[i][2] <= work[2])

                    {

                        for (int j = 0; j < work.length; j++)

                        {

                            work[j] = work[j] + allocation[i][j];

                        }

                        System.out.print("P" + i + " ");

                        finish[i] = true;

                        t++;

                    }

                }

            }

            h++;

        }

        if (t == max.length)

        {

            System.out.println("\nThe process is safe");

        }

        else

        {

            System.out.println("\nnot safe");

        }

    }

}

OUTPUT



import java.util.\*;

public

class Exp4Q2

{

public

    static void main(String[] args)

    {

        int[][] allocation = {{0, 1, 1, 0}, {1, 2, 3, 1}, {1, 3, 6, 5}, {0, 6, 3, 2}, {0, 0, 1, 4}};

        int[][] max = {{0, 2, 1, 0}, {1, 6, 5, 2}, {2, 3, 6, 6}, {0, 6, 5, 2}, {0, 6, 5, 6}};

        int[] available = {1, 5, 2, 0};

        int[][] need = new int[5][4];

        for (int i = 0; i < allocation.length; i++)

        {

            for (int j = 0; j < available.length; j++)

            {

                need[i][j] = max[i][j] - allocation[i][j];

            }

        }

        int[] work = available;

        boolean[] finish = new boolean[max.length];

        for (int i = 0; i < max.length; i++)

        {

            finish[i] = false;

        }

        int h = 0;

        int t = 0;

        System.out.println("Sequence");

        while (h <= 5)

        {

            for (int i = 0; i < max.length; i++)

            {

                if (finish[i] == false)

                {

                    if (need[i][0] <= work[0] && need[i][1] <= work[1] && need[i][2] <= work[2] && need[i][3] <= work[3])

                    {

                        for (int j = 0; j < work.length; j++)

                        {

                            work[j] = work[j] + allocation[i][j];

                        }

                        System.out.print("P" + i + " ");

                        finish[i] = true;

                        t++;

                    }

                }

            }

            h++;

        }

        if (t == max.length)

        {

            System.out.println("\nThe process is safe");

        }

        else

        {

            System.out.println("\n not safe");

        }

    }

OUTPUT:

