**CC LAB 1-3**

**Q1:** Find the makespan for the following ETC matrix having 7 task and 3 machine using min-min algorithm. Find the time complexity of min-min algorithm.

**Ans:**

#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

int main(){

    int n,m;

    printf("Enter the number of virtual machines and number of tasks: ");

    scanf("%d %d",&n,&m);

    int a[m][n];

    bool visited[m];

    for(int i=0;i<m;i++){

        printf("Enter times for task %d\n",i+1);

        for(int j=0;j<n;j++){

            scanf("%d",&a[i][j]);

        }

        visited[i]=false;

    }

    int cnt=0;

    int max\_time[n];

    for(int i=0;i<n;i++){

        max\_time[i]=0;

    }

    while(cnt < m){

        int min=1e9;

        int machine\_no=-1;

        int task\_no=-1;

        for(int i=0;i<m;i++){

            if(visited[i]==true)continue;

            for(int j=0;j<n;j++){

                if(min> a[i][j]){

                    min=a[i][j];

                    machine\_no=j;

                    task\_no=i;

                }

            }

        }

        visited[task\_no]=true;

        max\_time[machine\_no]=min;

        cnt++;

        printf("Task %d is assigned to machine %d\n",task\_no+1,machine\_no+1);

        for(int i=0;i<m;i++){

            if(visited[i]==true)continue;

            a[i][machine\_no]+=min;

        }

    }

    int ans=0;

    for(int i=0;i<n;i++){

            if(ans<max\_time[i]){

                ans=max\_time[i];

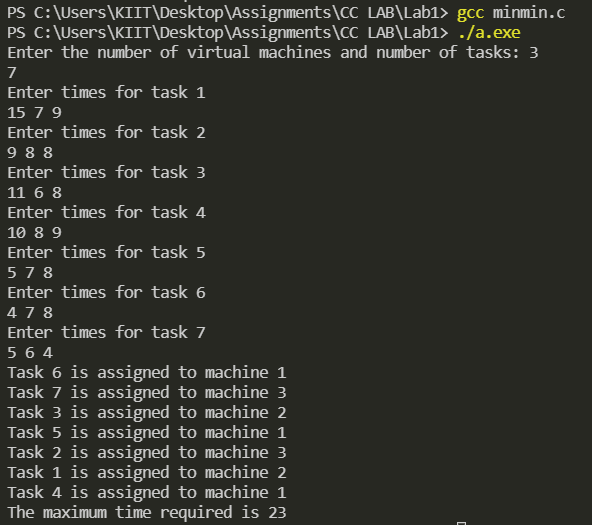
            }

    }

    printf("The maximum time required is %d\n",ans);

}

OUTPUT:



**Q2:** Find the makespan for the following ETC matrix having 7 task and 3 machine using max-min algorithm. Find the time complexity of max-min algorithm.

**Ans:**

#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

int main(){

    int n,m;

    printf("Enter the number of virtual machines and number of tasks: ");

    scanf("%d %d",&n,&m);

    int a[m][n];

    bool visited[m];

    for(int i=0;i<m;i++){

        printf("Enter times for task %d\n",i+1);

        for(int j=0;j<n;j++){

            scanf("%d",&a[i][j]);

        }

        visited[i]=false;

    }

    int cnt=0;

    int max\_time[n];

    for(int i=0;i<n;i++){

        max\_time[i]=0;

    }

    while(cnt < m){

        int machine\_no=-1;

        int task\_no=-1;

        int check\_time[m];

        int check\_machine[m];

        for(int i=0;i<m;i++){

            if(visited[i]==true){

                check\_time[i]=0;

                continue;

            }

            int min=1e9;

            for(int j=0;j<n;j++){

                if(min> a[i][j]){

                    min=a[i][j];

                    check\_machine[i]=j;

                }

            }

            check\_time[i]=min;

        }

        int min=0;

        for(int i=0;i<m;i++){

            if(min< check\_time[i]){

                min=check\_time[i];

                machine\_no=check\_machine[i];

                task\_no=i;

            }

        }

        visited[task\_no]=true;

        max\_time[machine\_no]=min;

        cnt++;

        printf("Task %d is assigned to machine %d and time is %d\n", task\_no+1,machine\_no+1,min);

        for(int i=0;i<m;i++){

            if(visited[i]==true)continue;

            a[i][machine\_no]+=min;

        }

    }

    int ans=0;

    for(int i=0;i<n;i++){

            if(ans<max\_time[i]){

                ans=max\_time[i];

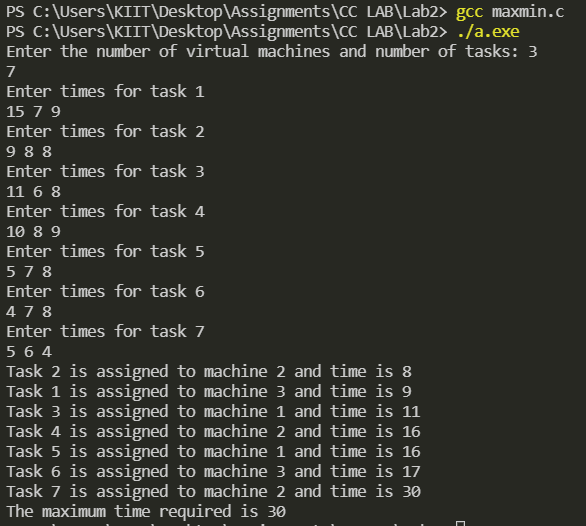
            }

    }

    printf("The maximum time required is %d\n",ans);

}

**OUTPUT:**

****