Panipat Institute of Engineering & Technology,
Samalkha (Haryana)
Computer Science & Engineering Department

Operating Systems Lab

PC-CS212AL

Submitted to:
Ms. Deepti Dhingra
(Assistant Professor CSE)

Submitted by:
Anmol Baranwal
2820208
B.Tech CSE 4th Sem A3

Affiliated to

Kurukshetra University Kurukshetra, India

INDEX

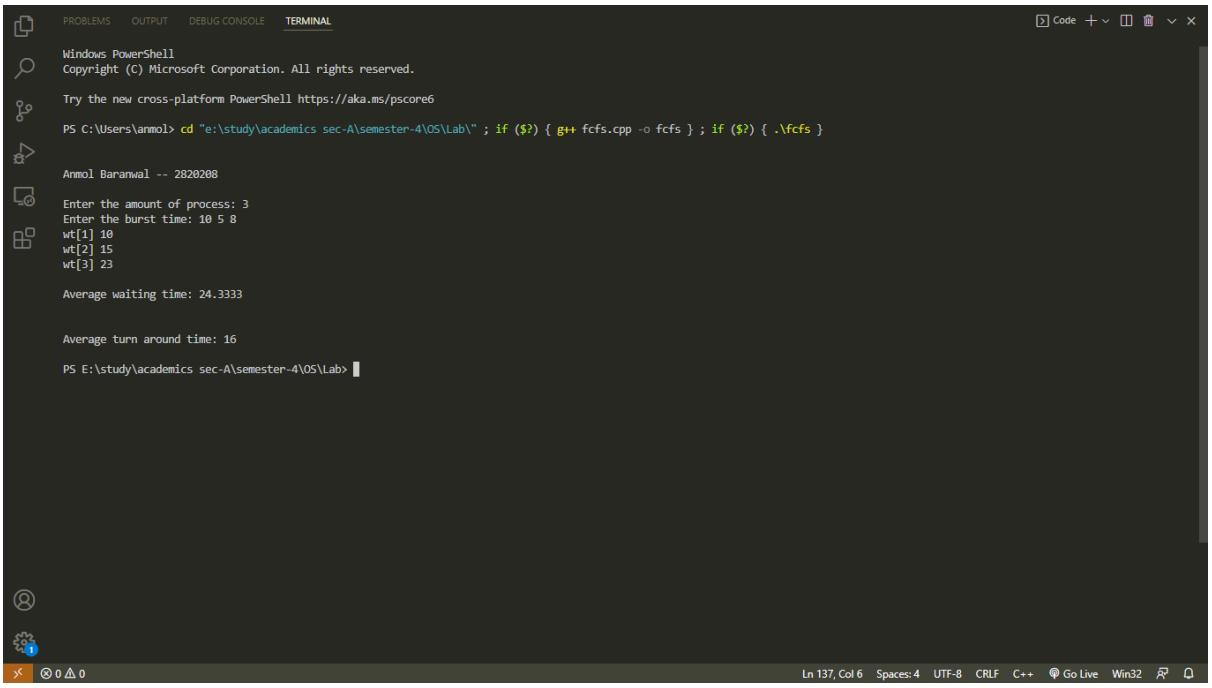
INDEX

Experiment 1.
Simulation of the CPU scheduling algorithm - FCFS

Program:-

```
#include<bits/stdc++.h>
using namespace std;

int main()
{
   int n,sumTAT=0,sumWT=0;

   cout<<"\n\n";
   cout<<"Anmol Baranwal -- 2820208";
   cout<<"\n\n";

   cout<<"Enter the amount of process: ";
   cin>>n;
   int arrivalTime[] {0};
   int wt[n], TAT[n], bt[n], AT[0];

   cout<<"Enter the burst time: ";
   for(size_t i=0;i<n;i++){
      cin>>bt[i];
   }

   wt[0]=0;
   for(size_t i=1;i<=n;i++){
      wt[i]=wt[i-1]+bt[i-1];
      cout<<"wt["<<i<<"] "<<wt[i]<<"  \n";
      sumWT+=wt[i];
   }
   for(int i=0;i<n;i++){
      TAT[i]=wt[i]+bt[i];
      sumTAT+=TAT[i];
      sumWT+=wt[i];
   }
   double avgTAT= ((double)sumTAT)/n;
   double avgWT= ((double)sumWT)/n;

   cout<<"\nAverage waiting time: "<<avgWT<<endl<<endl;
   cout<<"\nAverage turn around time: "<<avgTAT<<endl<<endl;

   return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\" ; if ($?) { g++ fcfs.cpp -o fcfs } ; if ($?) { .\fcfs }
Anmol Baranwal -- 2820208
Enter the amount of process: 3
Enter the burst time: 10 5 8
wt[1] 10
wt[2] 15
wt[3] 23
Average waiting time: 24.3333
Average turn around time: 16
PS E:\study\academics sec-A\semester-4\OS\Lab>
Ln 137, Col 6
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32

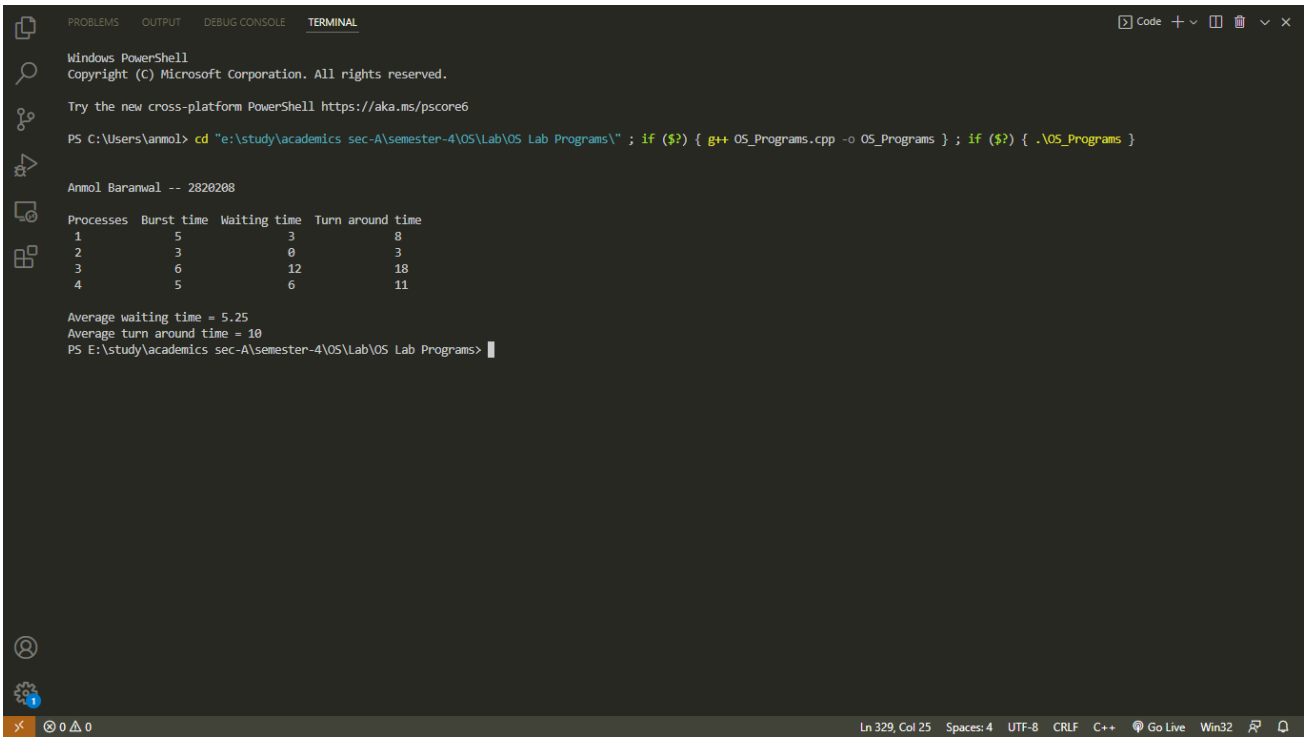
Experiment 2.
Simulation of the CPU scheduling algorithm - SJF

Program:-

```
#include <bits/stdc++.h>
using namespace std;

struct Process {
  int pid; // Process ID
  int bt; // Burst Time
  int art; // Arrival Time
};
void findTurnAroundTime(Process proc[], int n, int wt[], int tat[]) {
  for (int i = 0; i < n; i++)
  tat[i] = proc[i].bt + wt[i];
}

void findWaitingTime(Process proc[], int n, int wt[]) {
  int rt[n];
  for (int i = 0; i < n; i++)
  rt[i] = proc[i].bt;
  int complete = 0, t = 0, minm = INT_MAX;
  int shortest = 0, finish_time;
  bool check = false;
  while (complete != n) {
    for (int j = 0; j < n; j++) {
      if ((proc[j].art <= t) && (rt[j] < minm) && rt[j] > 0) {
        minm = rt[j];
        shortest = j;
        check = true;
      }
    }
    if (check == false) {
      t++;
      continue;
    }

    rt[shortest]--;
    minm = rt[shortest];
    if (minm == 0)
      minm = INT_MAX;

      if (rt[shortest] == 0) {
        complete++;
```

```
            check = false;
            finish_time = t + 1;
            // Calculate waiting time
            wt[shortest] = finish_time -
            proc[shortest].bt -
            proc[shortest].art;
            if (wt[shortest] < 0)
               wt[shortest] = 0;
         }
         t++;
   }
}

void findavgTime(Process proc[], int n) {
   int wt[n], tat[n], total_wt = 0,
   total_tat = 0;

   findWaitingTime(proc, n, wt);

   findTurnAroundTime(proc, n, wt, tat);
   cout << "Processes " << " Burst time " << " Waiting time " << " Turn around time\n";
   for (int i = 0; i < n; i++) {
      total_wt = total_wt + wt[i];
      total_tat = total_tat + tat[i];
      cout << " " << proc[i].pid << "\t\t" << proc[i].bt << "\t\t " << wt[i] << "\t\t " << tat[i] << endl;
   }
   cout << "\nAverage waiting time = " << (float)total_wt / (float)n; cout << "\nAverage turn around
time = " << (float)total_tat / (float)n;
}
int main() {

   cout<<"\n\nAnmol Baranwal -- 2820208\n\n";
   Process proc[] = { { 1, 5, 1 }, { 2, 3, 1 }, { 3, 6, 2 }, { 4, 5, 3 } };
   int n = sizeof(proc) / sizeof(proc[0]);
   findavgTime(proc, n);
   return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal -- 2820208
Processes Burst time Waiting time Turn around time
1 5 3 8
2 3 0 3
3 6 12 18
4 5 6 11
Average waiting time = 5.25
Average turn around time = 10
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
Ln 329, Col 25
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32

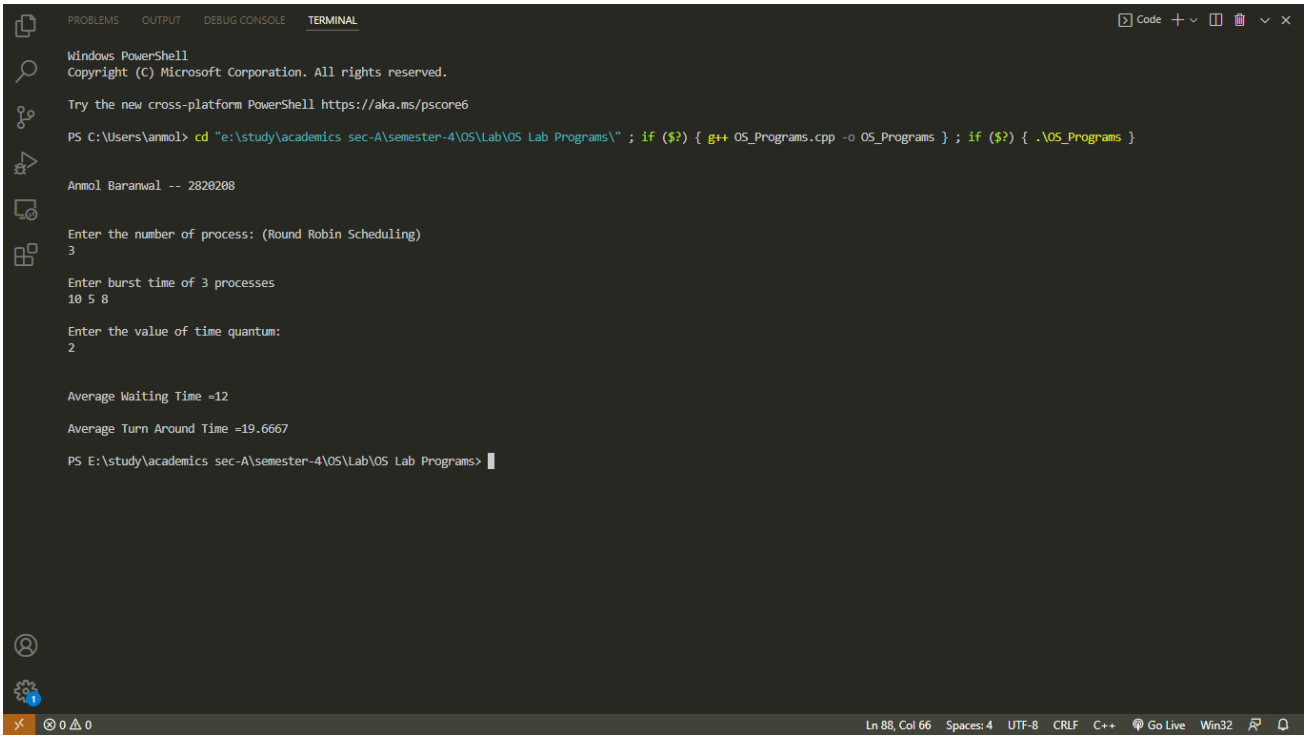
Experiment 3.
Simulation of the CPU scheduling algorithm – Round Robin

Program:-

```
#include<bits/stdc++.h>
using namespace std;

void findWaitingTime(int processes[] ,int burstTime[], int WT[], int quantum, int n)
{
   int remBT[20] {0};
   int t=0;

   for(int i=0;i<n;i++){
      remBT[i]=burstTime[i];
   }

   while(1){

      bool finish =true;

      for(int i=0;i<n;i++){
         if(remBT[i] > 0){

            finish = false;  // the process has some burst time so it has not finished yet

            if(remBT[i] > quantum){
               t += quantum;
               remBT[i] -=quantum;
            }
            else{
               t += remBT[i];
               WT[i] = t - burstTime[i];
               remBT[i]=0;
            }
         }
      }
      if(finish==true)
         break;
   }
}
void findTurnAroundTime(int processes[], int WT[], int burstTime[], int TAT[], int n)
{
   for(int i=0;i<n;i++){
      TAT[i] = WT[i] + burstTime[i];
   }
}
```

```
void findAvgTime(int processes[], int burstTime[], int quantum, int n)
{
   int WT[20] {0}, TAT[20] {0}, total_TAT=0, total_WT=0;

   // find waiting time of all the process
   findWaitingTime(processes, burstTime, WT, quantum, n);

   // find turn around time of all the process
   findTurnAroundTime(processes, WT, burstTime, TAT, n);


   for(int i=0;i<n;i++){
      total_WT += WT[i];
      total_TAT += TAT[i];
   }
   cout<<"\n\nAverage Waiting Time ="<< (float)total_WT/(float)n<<"\n";
   cout<<"\nAverage Turn Around Time ="<< (float)total_TAT/(float)n<<"\n\n";
}

int main()
{
   int processes[20] {0}, n;

   cout<<"\n\n";
   cout<<"Anmol Baranwal -- 2820208";
   cout<<"\n\n";

   cout<<"\nEnter the number of process: \n";
   cin>>n;

   // int AT[n] {0}, BT[n] {0}, n, quantum=1, WT[n] {0}, TAT[n] {0}, remBT[n] {0};
   // int sumTAT=0, sumWT=0;

   int burstTime[20] {0}, quantum=0;

   cout<<"\nEnter burst time of "<<n<<" processes\n";

   for(int i=0;i<n;i++){
      processes[i]=i+1;
      cin>>burstTime[i];
   }

   cout<<"\nEnter the value of time quantum: \n";
   cin>>quantum;

   findAvgTime(processes, burstTime, quantum, n);
   return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal -- 2820208
Enter the number of process: (Round Robin Scheduling)
3
Enter burst time of 3 processes
10 5 8
Enter the value of time quantum:
2
Average Waiting Time =12
Average Turn Around Time =19.6667
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 88, Col 66
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32

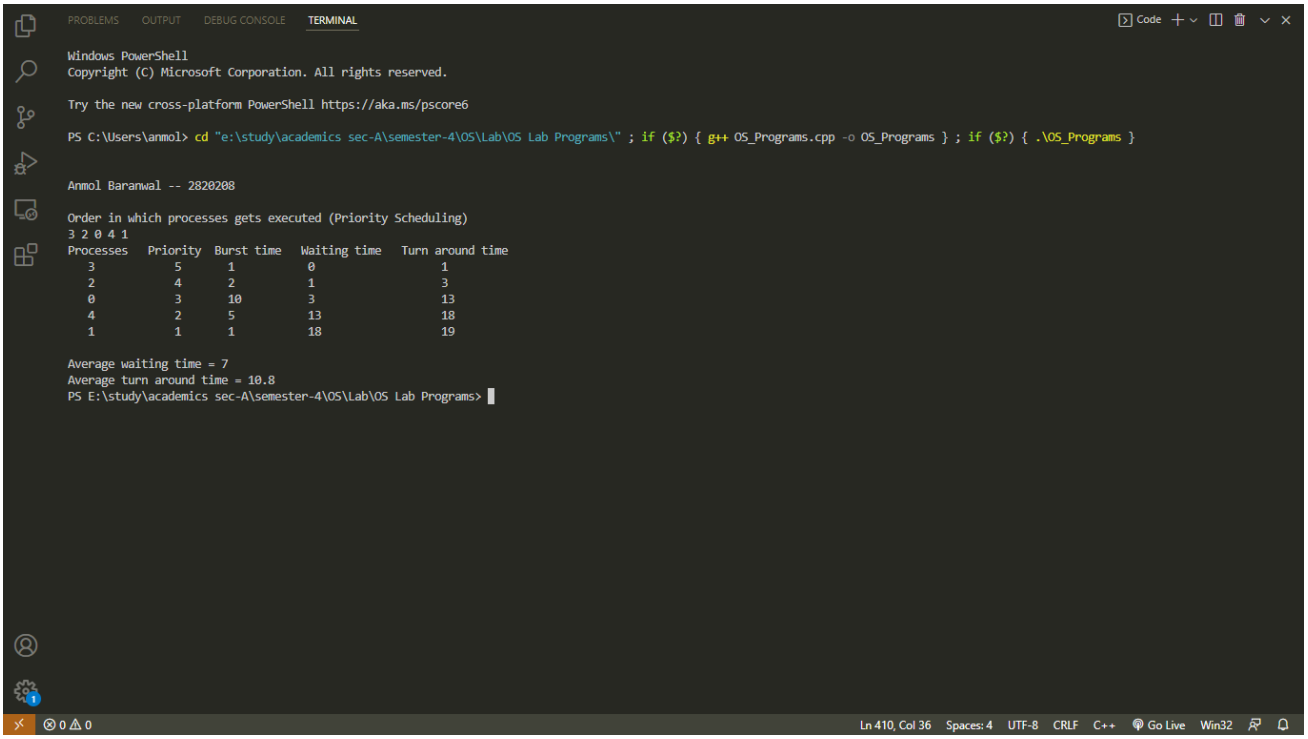
Experiment 4.
Simulation of the CPU scheduling algorithm - Priority

Program:-

```
#include<bits/stdc++.h>
using namespace std;

struct Process
{
   int pid;  // Process ID
   int bt;   // CPU Burst time required
   int priority; // Priority of this process
};

bool comparison(Process a, Process b)
{
   return (a.priority > b.priority);
}


void findWaitingTime(Process proc[], int n,
                     int wt[])
{

   wt[0] = 0;

   for (int  i = 1; i < n ; i++ )
      wt[i] =  proc[i-1].bt + wt[i-1] ;
}

void findTurnAroundTime( Process proc[], int n,
                         int wt[], int tat[])
{

   for (int  i = 0; i < n ; i++)
      tat[i] = proc[i].bt + wt[i];
}

void findavgTime(Process proc[], int n)
{
   int wt[n], tat[n], total_wt = 0, total_tat = 0;

   findWaitingTime(proc, n, wt);

   findTurnAroundTime(proc, n, wt, tat);
```

```
    cout << "\nProcesses  "<< " Priority " << " Burst time  "
         << " Waiting time  " << " Turn around time\n";

    for (int  i=0; i<n; i++)
    {
        total_wt = total_wt + wt[i];
        total_tat = total_tat + tat[i];
        cout << "   " << proc[i].pid << "\t\t"
             << proc[i].priority <<" \t"
             << proc[i].bt << "  \t    " << wt[i]
             << "\t\t\t" << tat[i] <<endl;
    }

    cout << "\nAverage waiting time = "
         << (float)total_wt / (float)n;
    cout << "\nAverage turn around time = "
         << (float)total_tat / (float)n;
}

void priorityScheduling(Process proc[], int n)
{

    sort(proc, proc + n, comparison);

    cout<< "Order in which processes gets executed (Priority Scheduling) \n";
    for (int  i = 0 ; i <  n; i++)
        cout << proc[i].pid <<" " ;

    findavgTime(proc, n);
}

int main()
{

    cout<<"\n\nAnmol Baranwal -- 2820208\n\n";
    Process proc[] = {{0, 10, 3}, {1, 1, 1}, {2, 2, 4}, {3, 1, 5}, {4, 5, 2}};
    int n = sizeof proc / sizeof proc[0];
    priorityScheduling(proc, n);

    return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal -- 2820208
Order in which processes gets executed (Priority Scheduling)
3 2 0 4 1
Processes Priority Burst time Waiting time Turn around time
3 5 1 0 1
2 4 2 1 3
0 3 10 3 13
4 2 5 13 18
1 1 1 18 19
Average waiting time = 7
Average turn around time = 10.8
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 410, Col 36
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32

Experiment 5.
Write a program to implement Banker's Algorithm

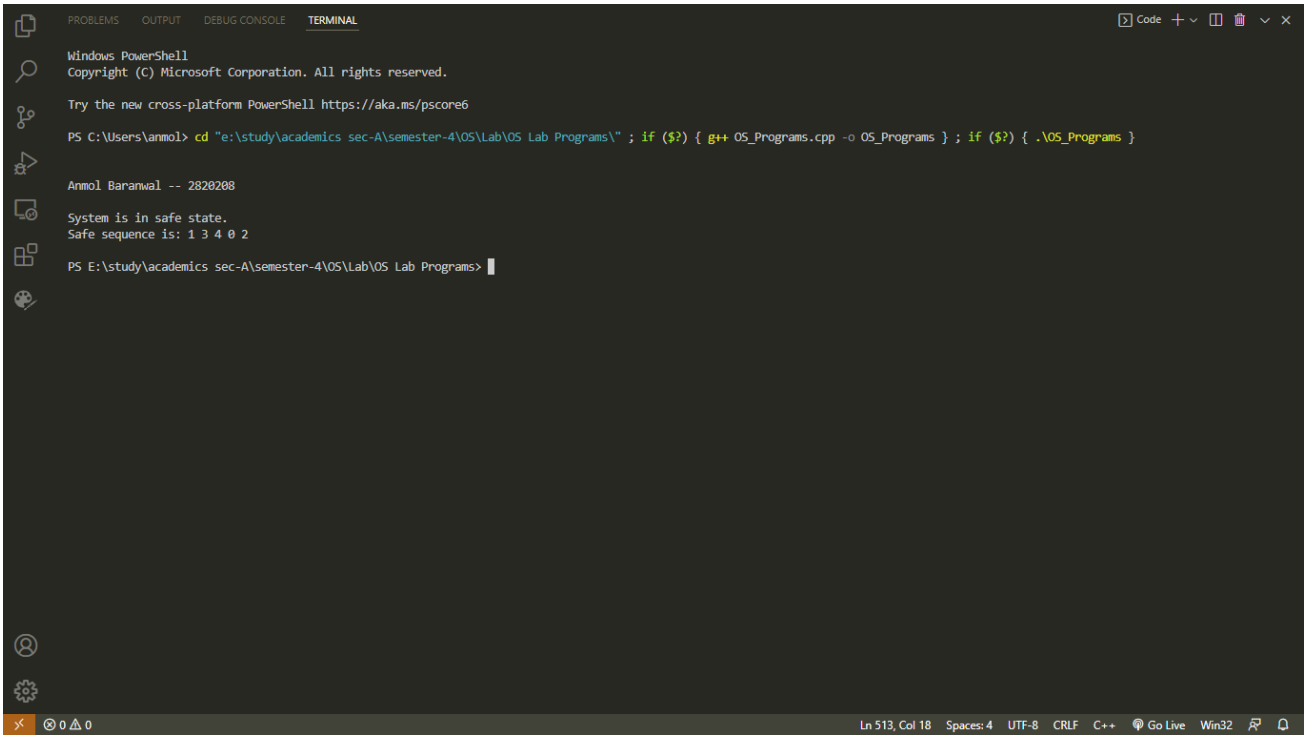
Program:-

```
#include<iostream>
using namespace std;

// Number of processes
const int P = 5;

// Number of resources
const int R = 3;

void calculateNeed(int need[P][R], int maxm[P][R],
       int allot[P][R])
{
  for (int i = 0 ; i < P ; i++)
    for (int j = 0 ; j < R ; j++)
      need[i][j] = maxm[i][j] - allot[i][j];
}

bool isSafe(int processes[], int avail[], int maxm[][R],
     int allot[][R])
{
  int need[P][R];
  calculateNeed(need, maxm, allot);
  bool finish[P] = {0};
  int safeSeq[P];
  int work[R];

  for (int i = 0; i < R ; i++)
    work[i] = avail[i];

  int count = 0;
  while (count < P)
  {
    bool found = false;
    for (int p = 0; p < P; p++)
    {
      if (finish[p] == 0)
      {
        int j;
        for (j = 0; j < R; j++)
          if (need[p][j] > work[j])
            break;
        if (j == R)
        {
```

```
        for (int k = 0 ; k < R ; k++)
          work[k] += allot[p][k];

        safeSeq[count++] = p;
        finish[p] = 1;
        found = true;
       }
      }
    }

    if (found == false){
      cout << "System is not in safe state";
      return false;
    }
  }
  cout << "System is in safe state.\nSafe"
    " sequence is: ";
  for (int i = 0; i < P ; i++)
    cout << safeSeq[i] << " ";

  return true;
}

int main()
{
    cout<<"\n\nAnmol Baranwal -- 2820208\n\n";
  int processes[] = {0, 1, 2, 3, 4};

  // Available instances of resources
  int avail[] = {3, 3, 2};

  // Maximum R that can be allocated
  int maxm[][R] = {{7, 5, 3},
        {3, 2, 2},
        {9, 0, 2},
        {2, 2, 2},
        {4, 3, 3}};

  // Resources allocated to processes
  int allot[][R] = {{0, 1, 0},
        {2, 0, 0},
        {3, 0, 2},
        {2, 1, 1},
        {0, 0, 2}};

  isSafe(processes, avail, maxm, allot);
    cout<<"\n\n";
  return 0;
}
```

```
int main() {
   cout<<"\n\nAnmol Baranwal -- 2820208\n\n";
   char job[10][10];
   int time[10], avail, tem[10], temp[10];
   int safe[10];
   int ind = 1, i, j, q, n, t;
   cout<<"Enter no of jobs: ";
   cin>>n;
   for (i = 0; i < n; i++) {
      cout<<"Enter name and time: ";
      cin>>job[i]>>time[i];
   }
   cout<<"Enter the available resources:";
   cin>>avail;
   cout<<"\n";
   for (i = 0; i < n; i++) {
      temp[i] = time[i];
      tem[i] = i;
   }
   for (i = 0; i < n; i++)
      for (j = i + 1; j < n; j++) {
         if (temp[i] > temp[j]) {
            t = temp[i];
            temp[i] = temp[j];
            temp[j] = t;
            t = tem[i];
            tem[i] = tem[j];
            tem[j] = t;
         }
      }
   for (i = 0; i < n; i++) {
      q = tem[i];
      if (time[q] <= avail) {
         safe[ind] = tem[i];
         avail = avail - tem[q];
         cout<<job[safe[ind]];
         ind++;
      } else {
         cout<<"No safe sequence\n";
      }
   }
   cout<<"\nSafe sequence is:";
   for (i = 1; i < ind; i++)
      cout<<"\n"<< job[safe[i]]<<" "<< time[safe[i]]<<",";

   return 0;
}
```

OUTPUT:

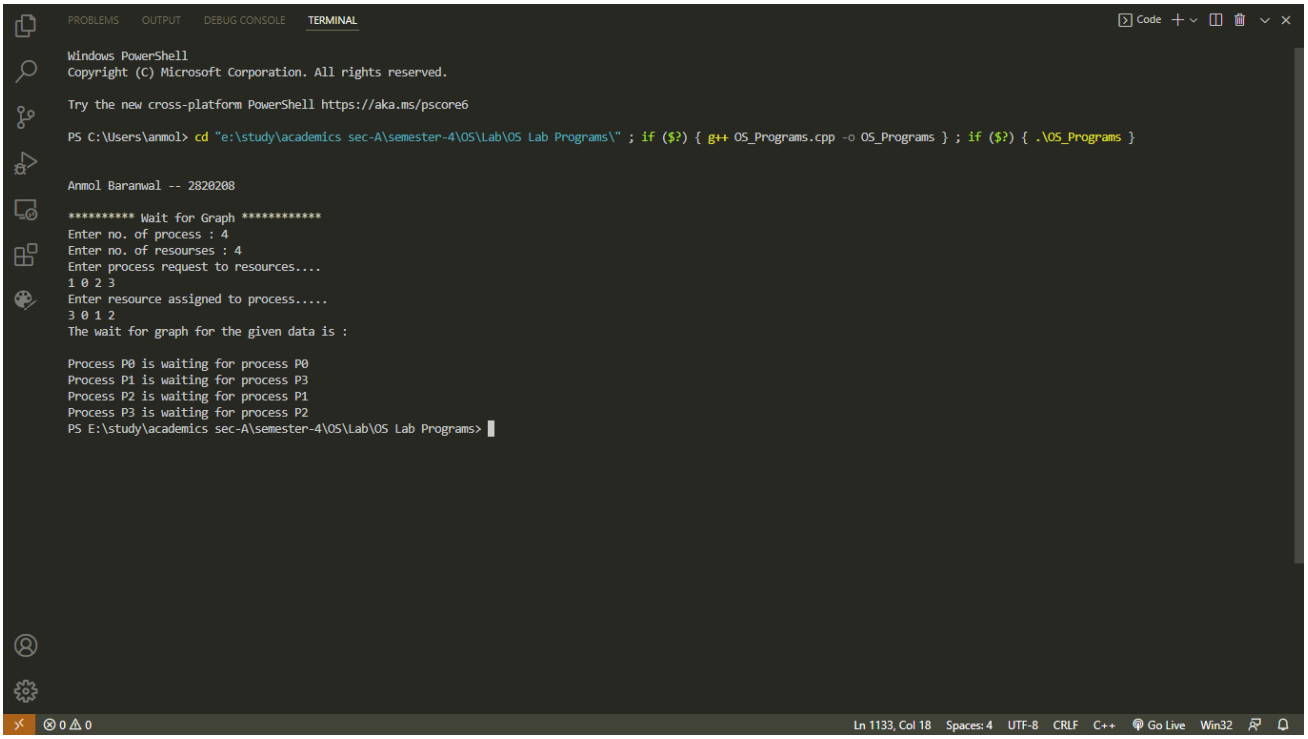

PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal -- 2820208
System is in safe state.
Safe sequence is: 1 3 4 0 2
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 513, Col 18
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32

Experiment 6.
Write a program to implement wait-for-graph

Program:-

```
#include<bits/stdc++.h>
using namespace std;

int main() {
   cout << "\n\nAnmol Baranwal -- 2820208\n\n";
   cout << "********** Wait for Graph **********\n";
   int p[10][2], r[10][2], wfg[10][2], i, j, k, n, m;
   cout<<"Enter no. of process : ";
   cin>>m;
   cout<<"Enter no. of resourses : ";
   cin>>n;
   cout<<"Enter process request to resources....\n";
   for (i = 0; i < m; i++) {
      cin>>p[i][1];
      if (p[i][1] < 0 || p[i][1] > n) {
         cout<<"Please enter valid resource(0-"<<n<<")\n";
         i--;
      }
   }
   cout<<"Enter resource assigned to process.....\n";
   for (i = 0; i < n; i++) {
      cin>>r[i][1];
   }
   for (i = 0; i < m; i++) {
      for (j = 0; j < n; j++) {
         if (p[i][1] == j) {
            for (k = 0; k < m; k++) {
               if (r[j][1] == k) {
                  wfg[i][1] = k;
                  k = m;
               }
            }
            j = n;
         }
      }
   }
   cout<<"The wait for graph for the given data is : \n";
   for (i = 0; i < m; i++) {
      if (wfg[i][1] >= 0 && wfg[i][1] < m) {
         cout<<"\nProcess P"<<i<<" is waiting for process P"<<wfg[i][1];
      }
   }
   return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal -- 2820208
********** Wait for Graph ***********
Enter no. of process : 4
Enter no. of resourses : 4
Enter process request to resources....
1 0 2 3
Enter resource assigned to process.....
3 0 1 2
The wait for graph for the given data is :
Process P0 is waiting for process P0
Process P1 is waiting for process P3
Process P2 is waiting for process P1
Process P3 is waiting for process P2
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 1133, Col 18
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32

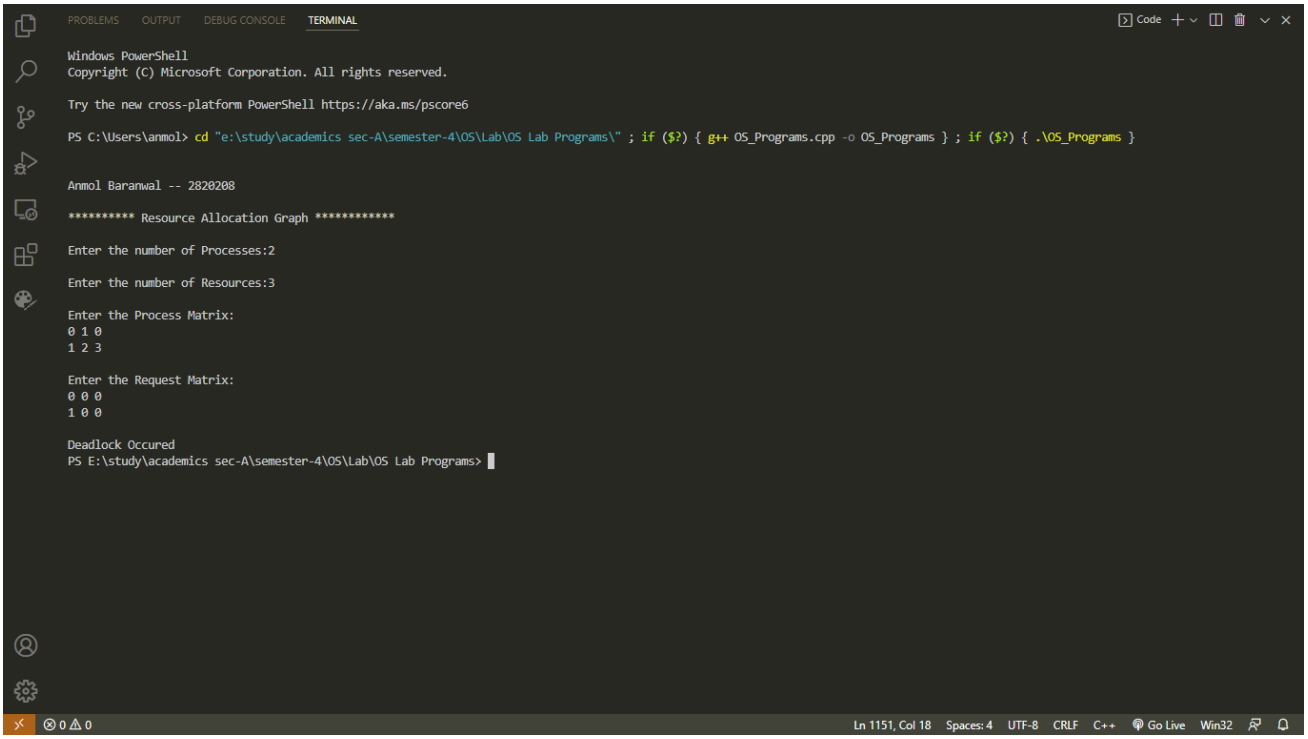
Experiment 7.
Write a program to implement Resource Allocation Group

Program:-

```
#include<bits/stdc++.h>
#include<conio.h>
using namespace std;

int proc, res, i, j, row = 0, flag = 0;
static int pro[3][3], req[3][3], st_req[3][3], st_pro[3][3];

int main() {
   cout << "\n\nAnmol Baranwal -- 2820208\n\n";
   cout << "********** Resource Allocation Graph **********\n";
   cout<<"\nEnter the number of Processes:";
   cin>>proc;
   cout<<"\nEnter the number of Resources:";
   cin>>res;

   cout<<"\nEnter the Process Matrix:";
   for (i = 0; i < proc; i++)
      for (j = 0; j < res; j++)
         cin>>pro[i][j];

   cout<<"\nEnter the Request Matrix:");
   for (i = 0; i < res; i++)
      for (j = 0; j < proc; j++)
         cin>>req[i][j];

   row = 0;
   while (!kbhit()) {
      for (i = 0; i < res; i++) {
         if (pro[row][i] == 1) {
            if (st_pro[row][i] > 1 && flag == 1) {
               cout<<"\nDeadlock Occured";
               return 0;
            }
            st_pro[row][i]++;
            row = i;
            break;
         }
      }

      for (i = 0; i < proc; i++) {
         if (req[row][i] == 1) {
            if (st_req[row][i] > 1) {
               cout<<"\nDeadlock Occured";
```

```
                return 0;
            }
            st_req[row][i]++;
            row = i;
            flag = 1;
            break;
        }
    }
  }
  cout<<"\nNo Deadlock Detected";

  return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal -- 2820208
********** Resource Allocation Graph ***********
Enter the number of Processes:2
Enter the number of Resources:3
Enter the Process Matrix:
0 1 0
1 2 3
Enter the Request Matrix:
0 0 0
1 0 0
Deadlock Occured
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 1151, Col 18
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32

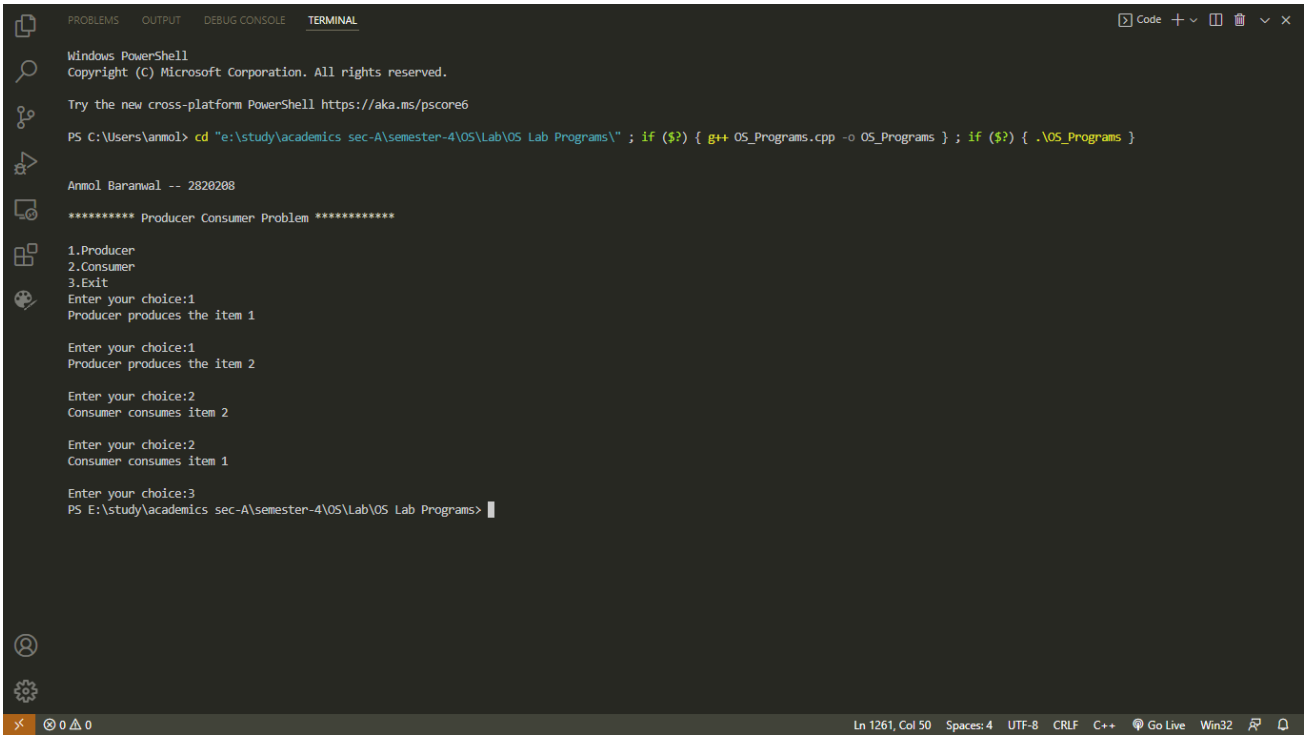
Experiment 8.
Write a program to implement Producer Consumer Problem

Program:-

```
#include<bits/stdc++.h>
using namespace std;

int mutex = 1, full = 0, empty = 3, x = 0;

int main() {
   cout << "\n\nAnmol Baranwal -- 2820208\n\n";
   cout << "********** Producer Consumer Problem **********\n";
   int n;
   void producer();
   void consumer();
   int wait(int);
   int signal(int);
   cout<<"\n1.Producer\n2.Consumer\n3.Exit";
   while (1) {
      cout<<"\nEnter your choice:";
      cin>>n;
      switch (n) {
         case 1:
            if ((mutex == 1) && (empty != 0))
               producer();
            else
               printf("Buffer is full!!");
            break;
         case 2:
            if ((mutex == 1) && (full != 0))
               consumer();
            else
               printf("Buffer is empty!!");
            break;
         case 3:
            return 0;
            break;
      }
   }
   return 0;
}

int wait(int s) {
   return (--s);
}

int signal(int s) {
```

```
    return (++s);
}

void producer() {
    mutex = wait(mutex);
    full = signal(full);
    empty = wait(empty);
    x++;
    cout<<"Producer produces the item "<<x<<endl;
    mutex = signal(mutex);
}

void consumer() {
    mutex = wait(mutex);
    full = wait(full);
    empty = signal(empty);
    cout<<"Consumer consumes item "<<x<<endl;
    x--;
    mutex = signal(mutex);
}
```

OUTPUT

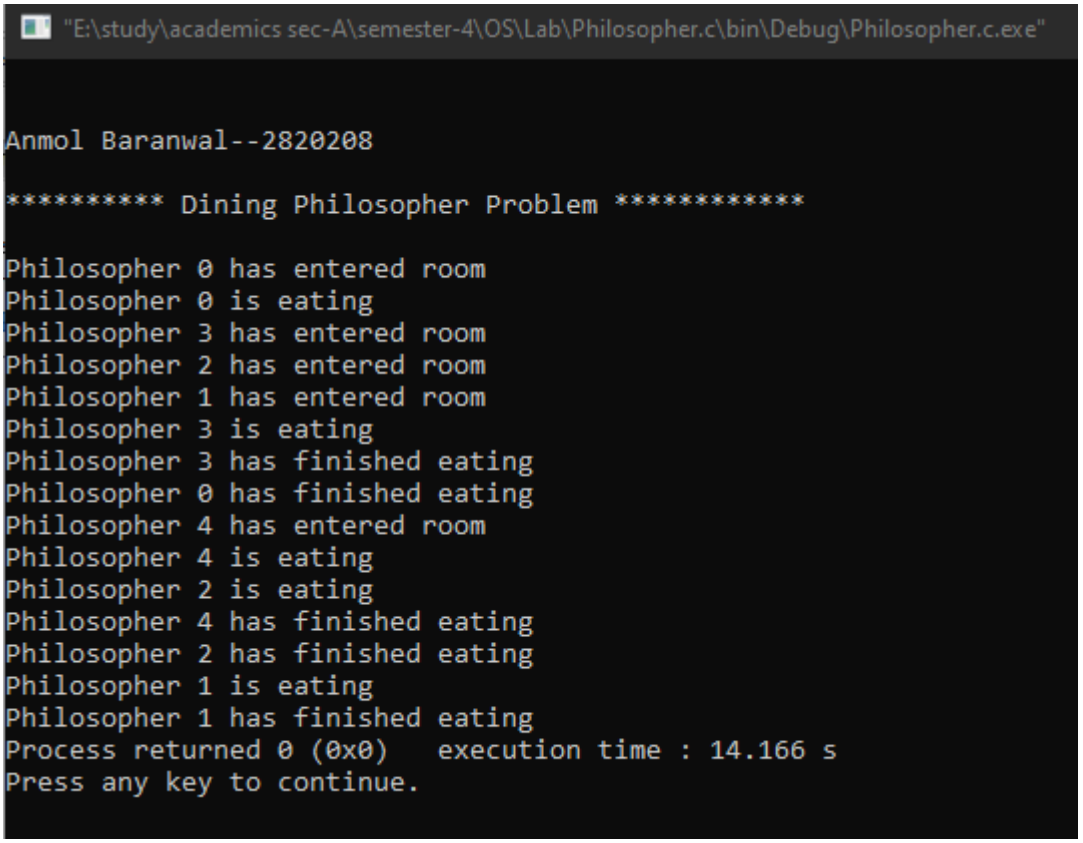

PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal -- 2820208
********** Producer Consumer Problem ***********
1.Producer
2.Consumer
3.Exit
Enter your choice:1
Producer produces the item 1
Enter your choice:1
Producer produces the item 2
Enter your choice:2
Consumer consumes item 2
Enter your choice:2
Consumer consumes item 1
Enter your choice:3
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 1261, Col 50
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32

Experiment 9.
Write a program to implement Dining Philosopher Problem

Program:-

```
#include<stdio.h>
#include<stdlib.h>
#include<pthread.h>
#include<semaphore.h>
#include<unistd.h>
sem_t room;
sem_t chopstick[5];
void * philosopher(void * );
void eat(int);
int main() {
   printf("\n\nAnmol Baranwal--2820208\n\n");
   printf("********** Dining Philosopher Problem ***********\n");
   int i, a[5];
   pthread_t tid[5];
   sem_init( & room, 0, 4);
   for (i = 0; i < 5; i++)
      sem_init( & chopstick[i], 0, 1);
   for (i = 0; i < 5; i++) {
      a[i] = i;
      pthread_create( & tid[i], NULL, philosopher, (void * ) & a[i]);
   }
   for (i = 0; i < 5; i++)
      pthread_join(tid[i], NULL);
}
void * philosopher(void * num) {
   int phil = * (int * ) num;
   sem_wait( & room);
   printf("\nPhilosopher %d has entered room", phil);
   sem_wait( & chopstick[phil]);
   sem_wait( & chopstick[(phil + 1) % 5]);
   eat(phil);
   sleep(2);
   printf("\nPhilosopher %d has finished eating", phil);
   sem_post( & chopstick[(phil + 1) % 5]);
   sem_post( & chopstick[phil]);
   sem_post( & room);
}
void eat(int phil) {
   printf("\nPhilosopher %d is eating", phil);
}
```

OUTPUT:

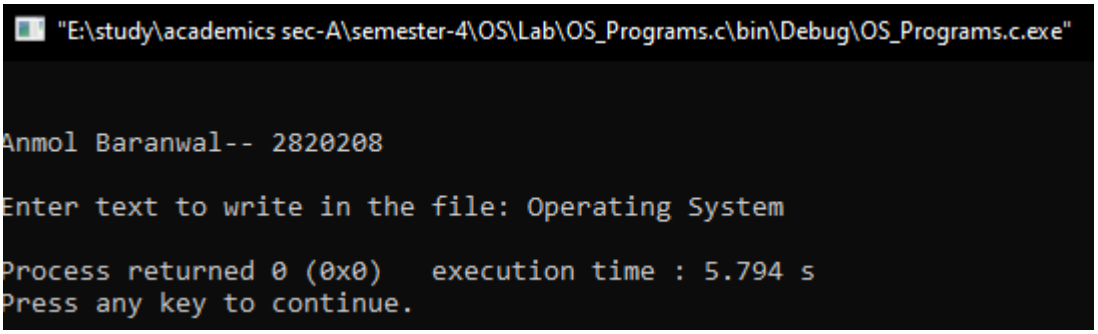

"E:\study\academics sec-A\semester-4\OS\Lab\Philosopher.c\bin\Debug\Philosopher.c.exe"
Anmol Baranwal--2820208
********** Dining Philosopher Problem *************
Philosopher 0 has entered room
Philosopher 0 is eating
Philosopher 3 has entered room
Philosopher 2 has entered room
Philosopher 1 has entered room
Philosopher 3 is eating
Philosopher 3 has finished eating
Philosopher 0 has finished eating
Philosopher 4 has entered room
Philosopher 4 is eating
Philosopher 2 is eating
Philosopher 4 has finished eating
Philosopher 2 has finished eating
Philosopher 1 is eating
Philosopher 1 has finished eating
Process returned 0 (0x0)   execution time : 14.166 s
Press any key to continue.

Experiment 10.
Write a program to implement System Calls

Program:-

```
#include<unistd.h>
#include<fcntl.h>
#include<sys/stat.h>
#include<sys/types.h>
#include<stdio.h>
int main()

{
   printf("\n\nAnmol Baranwal-- 2820208\n\n");
   int n,fd;
   char buff[50]; // declaring buffer
   printf("Enter text to write in the file: ");
   //read from keyboard, specifying 0 as fd for std input device
   //Here, n stores the number of characters
   n= read(0, buff, 50);
   // creating a new file using open.
   fd=open("file",O_CREAT | O_RDWR, 0777);
   //writting input data to file (fd)
   write(fd, buff, n);
   //Write to display (1 is standard fd for output device) write(1, buff, n);
   //closing the file
   int close(int fd);

   return 0;
}
```

OUTPUT:

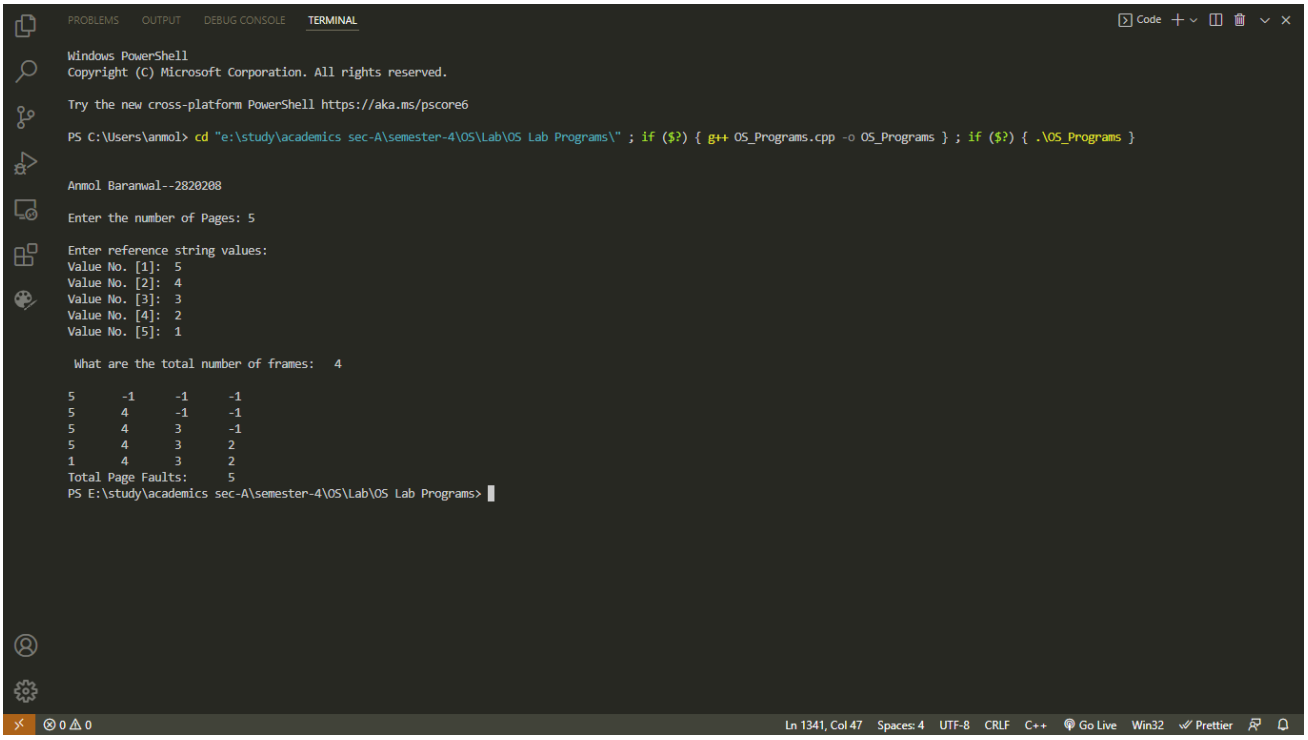

"E:\study\academics sec-A\semester-4\OS\Lab\OS_Programs.c\bin\Debug\OS_Programs.c.exe"
Anmol Baranwal-- 2820208
Enter text to write in the file: Operating System
Process returned 0 (0x0)   execution time : 5.794 s
Press any key to continue.

Experiment 11.a.
Write a Program for Page Replacement Algorithm - FIFO

Program:-

```
#include <stdio.h>

int main() {
   printf("\n\nAnmol Baranwal--2820208\n\n");
   int referenceString[10], pageFaults = 0, m, n, s, pages, frames;
   printf("Enter the number of Pages: ");
   scanf("%d", & pages);
   printf("\nEnter reference string values:\n");
   for (m = 0; m < pages; m++) {
      printf("Value No. [%d]:\t", m + 1);
      scanf("%d", & referenceString[m]);
   }
   printf("\n What are the total number of frames:\t"); {
      scanf("%d", & frames);
   }
   int temp[frames];
   for (m = 0; m < frames; m++) {
      temp[m] = -1;
   }
   for (m = 0; m < pages; m++) {
      s = 0;
      for (n = 0; n < frames; n++) {
         if (referenceString[m] == temp[n]) {
            s++;
            pageFaults--;
         }
      }
      pageFaults++;
      if ((pageFaults <= frames) && (s == 0)) {
         temp[m] = referenceString[m];
      } else if (s == 0) {
         temp[(pageFaults - 1) % frames] = referenceString[m];
      }
      printf("\n");
      for (n = 0; n < frames; n++) {
         printf("%d\t", temp[n]);
      }
   }
   printf("\nTotal Page Faults:\t%d\n", pageFaults);
   return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal--2820208
Enter the number of Pages: 5
Enter reference string values:
Value No. [1]: 5
Value No. [2]: 4
Value No. [3]: 3
Value No. [4]: 2
Value No. [5]: 1
What are the total number of frames: 4
5 -1 -1 -1
5 4 -1 -1
5 4 3 -1
5 4 3 2
1 4 3 2
Total Page Faults: 5
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 1341, Col 47
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32
Prettier

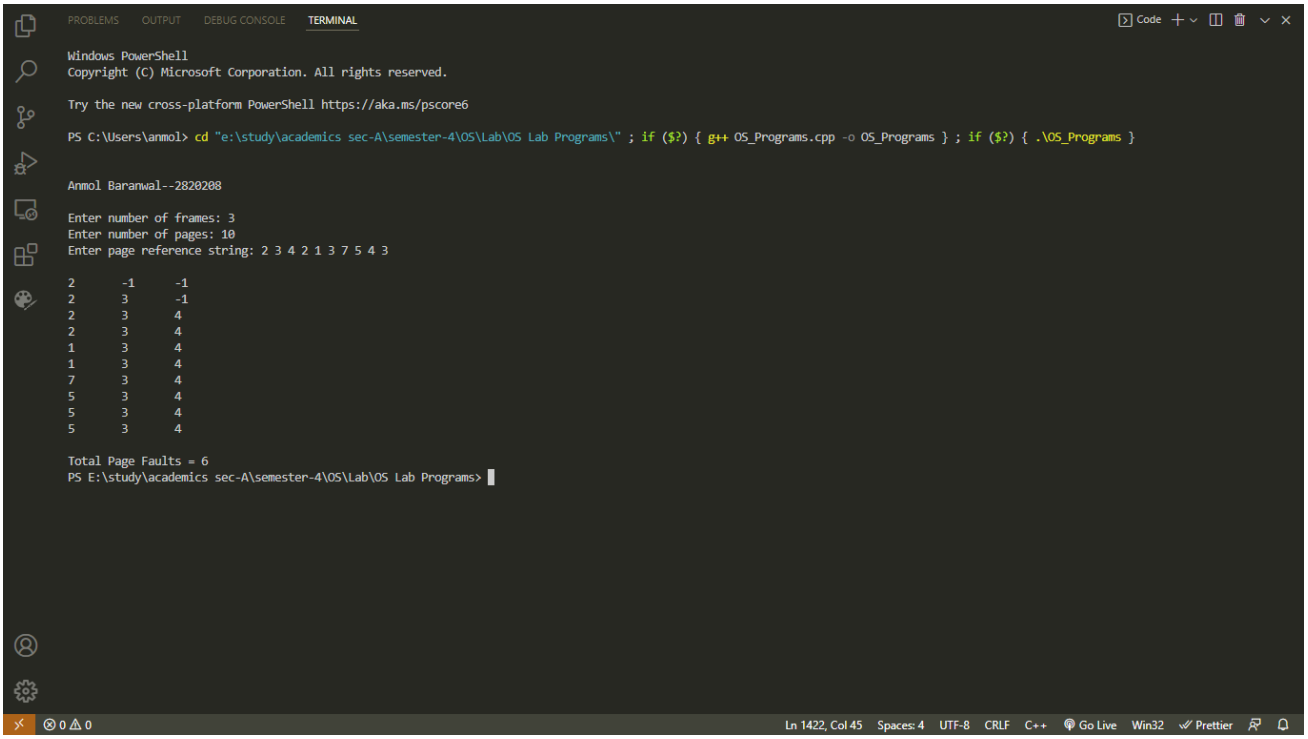
Experiment 11.b.
Write a Program for Optimal Page Replacement Algorithm

Program:-

```
#include<stdio.h>

int main() {
   printf("\n\nAnmol Baranwal--2820208\n\n");
   int no_of_frames, no_of_pages, frames[10], pages[30], temp[10], flag1, flag2, flag3, i, j, k, pos,
max, faults = 0;
   printf("Enter number of frames: ");
   scanf("%d", & no_of_frames);

   printf("Enter number of pages: ");
   scanf("%d", & no_of_pages);
   printf("Enter page reference string: ");
   for (i = 0; i < no_of_pages; ++i) {
      scanf("%d", & pages[i]);
   }

   for (i = 0; i < no_of_frames; ++i) {
      frames[i] = -1;
   }
   for (i = 0; i < no_of_pages; ++i) {
      flag1 = flag2 = 0;

      for (j = 0; j < no_of_frames; ++j) {
         if (frames[j] == pages[i]) {
            flag1 = flag2 = 1;
            break;
         }
      }
      if (flag1 == 0) {
         for (j = 0; j < no_of_frames; ++j) {
            if (frames[j] == -1) {
               faults++;
               frames[j] = pages[i];
               flag2 = 1;
               break;
            }
         }
      }
      if (flag2 == 0) {
         flag3 = 0;

         for (j = 0; j < no_of_frames; ++j) {
            temp[j] = -1;
```

```
            for (k = i + 1; k < no_of_pages; ++k) {
               if (frames[j] == pages[k]) {
                  temp[j] = k;
                  break;
               }
            }
         }

         for (j = 0; j < no_of_frames; ++j) {
            if (temp[j] == -1) {
               pos = j;
               flag3 = 1;
               break;
            }
         }

         if (flag3 == 0) {
            max = temp[0];
            pos = 0;

            for (j = 1; j < no_of_frames; ++j) {
               if (temp[j] > max) {
                  max = temp[j];


                  pos = j;
               }
            }
         }
         frames[pos] = pages[i];
         faults++;
      }

      printf("\n");

      for (j = 0; j < no_of_frames; ++j) {
         printf("%d\t", frames[j]);
      }
   }

   printf("\n\nTotal Page Faults = %d", faults);

   return 0;
}
```

OUTPUT:

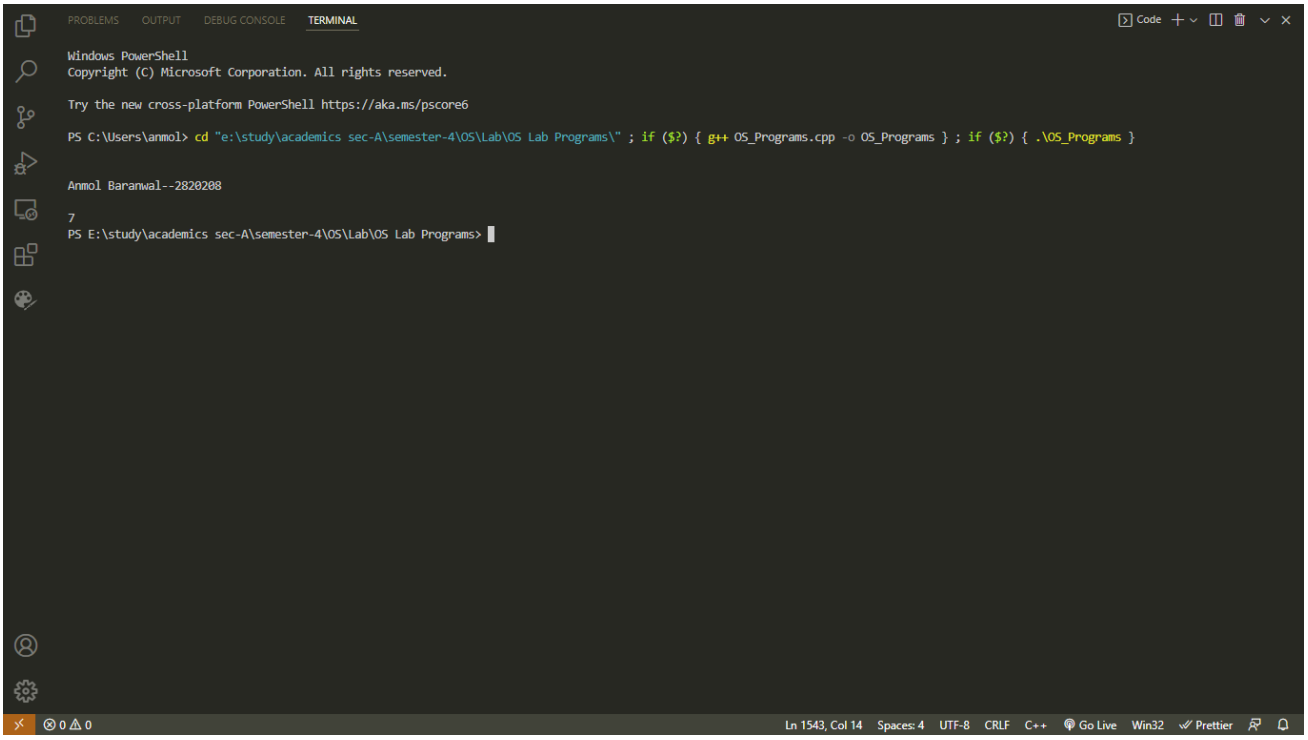

PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal--2820208
Enter number of frames: 3
Enter number of pages: 10
Enter page reference string: 2 3 4 2 1 3 7 5 4 3
2 -1 -1
2 3 -1
2 3 4
2 3 4
1 3 4
1 3 4
7 3 4
5 3 4
5 3 4
5 3 4
Total Page Faults = 6
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 1422, Col 45
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32
Prettier

Experiment 11.c.
Write a Program for Least Recently Used Page Replacement

Program:-

```
#include<bits/stdc++.h>
using namespace std;

int pageFaults(int pages[], int n, int capacity) {

   unordered_set < int > s;
   unordered_map < int, int > indexes;
   int page_faults = 0;
   for (int i = 0; i < n; i++) {
      if (s.size() < capacity) {
         if (s.find(pages[i]) == s.end()) {
            s.insert(pages[i]);
            page_faults++;
         }
      } else {
         indexes[pages[i]] = i;

         if (s.find(pages[i]) == s.end()) {
            int lru = INT_MAX, val;
            for (auto it = s.begin(); it != s.end(); it++) {
               if (indexes[ * it] < lru) {
                  lru = indexes[ * it];
                  val = * it;
               }
            }
            s.erase(val);
            s.insert(pages[i]);
            page_faults++;
         }
         indexes[pages[i]] = i;
      }
   }
   return page_faults;
}

int main() {
   cout << "\n\nAnmol Baranwal--2820208\n\n";
   int pages[] = {7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2};
   int n = sizeof(pages) / sizeof(pages[0]);
   int capacity = 4;
   cout << pageFaults(pages, n, capacity);
   return 0;
}
```

OUTPUT:

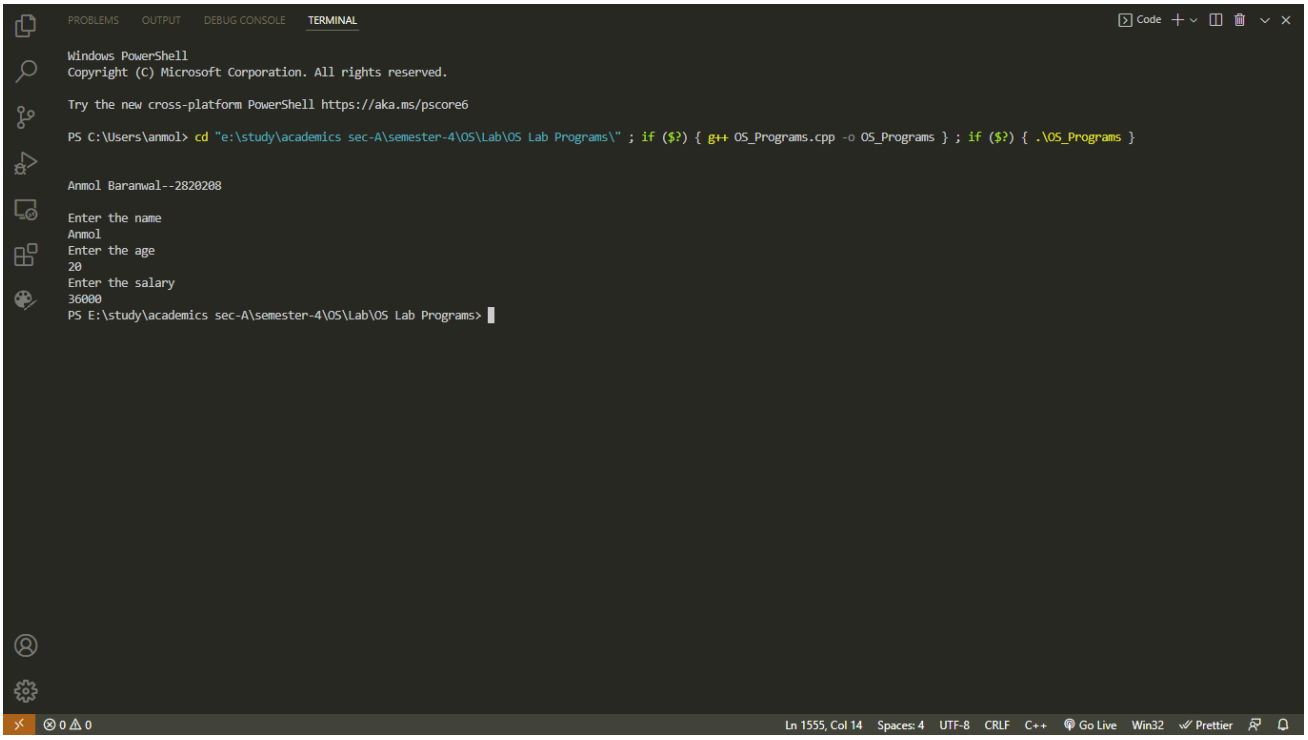

PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal--2820208
7
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 1543, Col 14
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32
Prettier

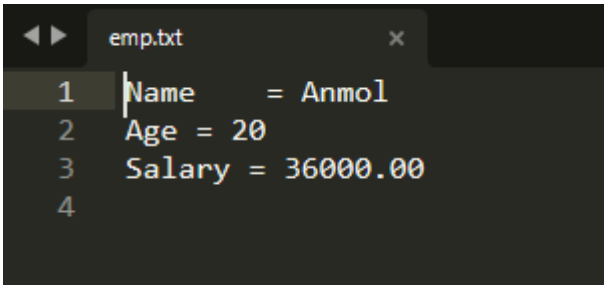
Experiment 12.
Write a program to implement File Operations

Program:-

```
#include <stdio.h>

int main() {
   printf("\n\nAnmol Baranwal--2820208\n\n");
   FILE
      *
      fptr;
   char name[20];
   int age;
   float salary;
   fptr = fopen("emp.txt", "w");
   /*open for writing*/
   if (fptr == NULL) {
      printf("File does not exists\n");
      return 0;
   }
   printf("Enter the name\n");
   scanf("%s", name);
   fprintf(fptr, "Name = %s\n", name);
   printf("Enter the age\n");
   scanf("%d", & age);
   fprintf(fptr, "Age  = %d\n", age);
   printf("Enter the salary\n");
   scanf("%f", & salary);
   fprintf(fptr, "Salary = %.2f\n", salary);
   fclose(fptr);

   return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal--2820208
Enter the name
Anmol
Enter the age
20
Enter the salary
36000
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
Ln 1555, Col 14
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32
Prettier

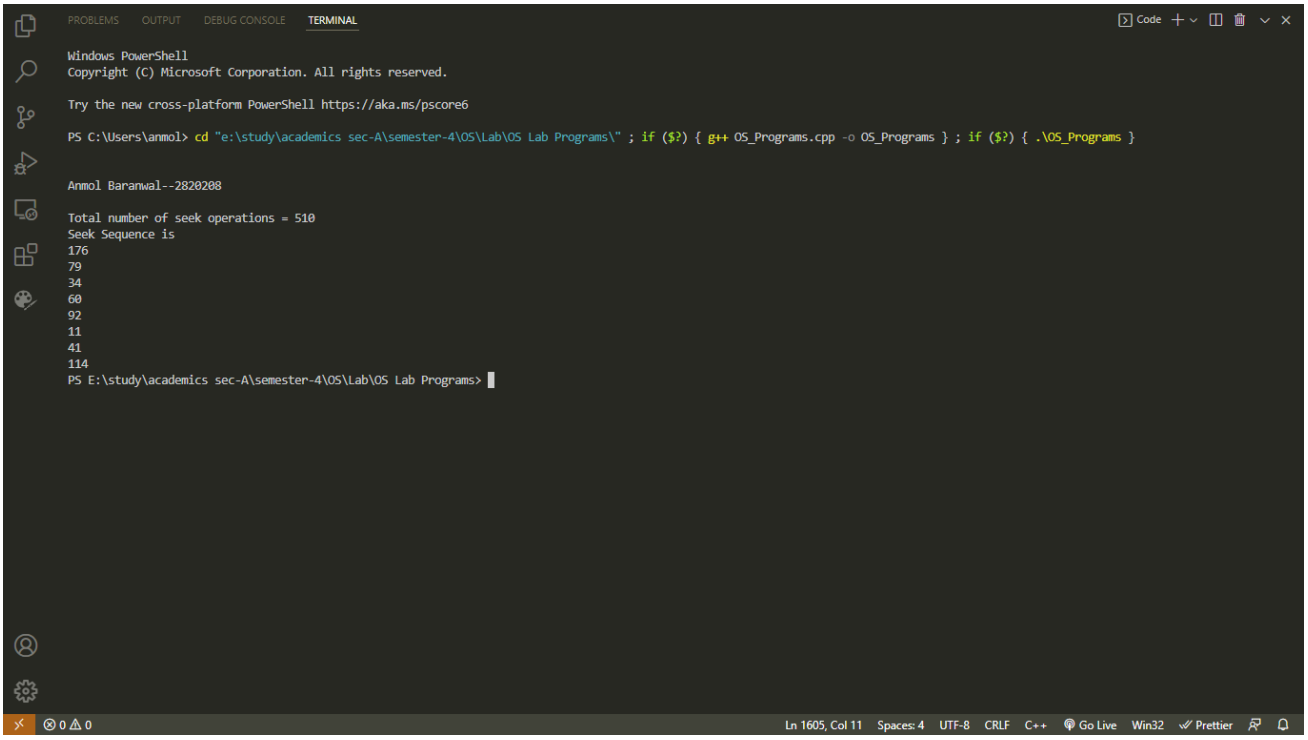

emp.txt
1 Name = Anmol
2 Age = 20
3 Salary = 36000.00
4

Experiment 13.a.
Write a program to implement FCFS Disk Scheduling Algorithm

Program:-

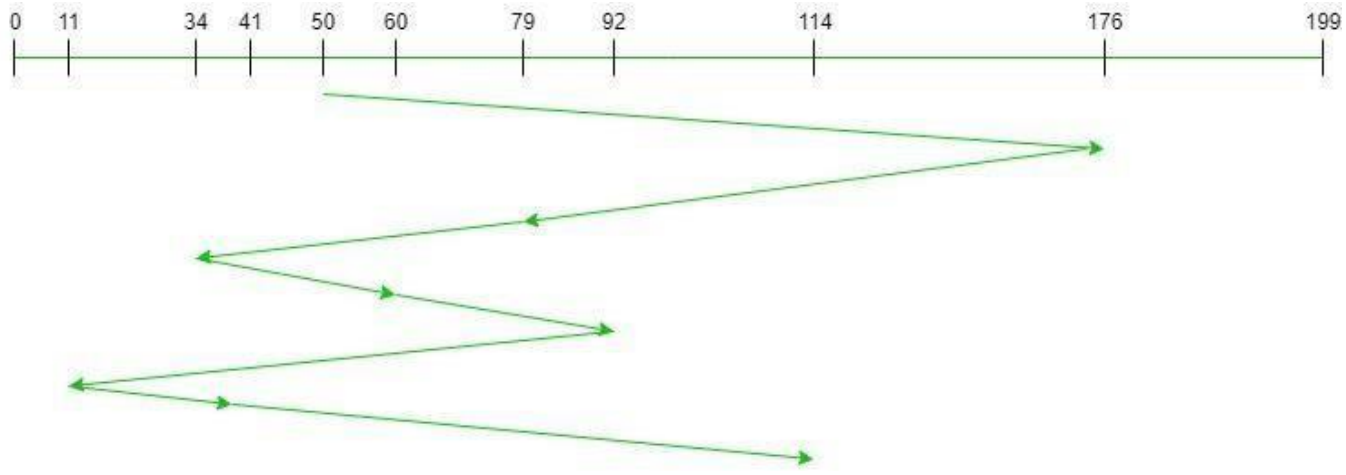
```
#include <bits/stdc++.h>
using namespace std;
int size = 8;

void FCFS(int arr[], int head)
{
   cout <<"\n\nAnmol Baranwal--2820208\n\n";
   int seek_count = 0;
   int distance, cur_track;
   for (int i = 0; i < size; i++) {
      cur_track = arr[i];
      distance = abs(cur_track - head);
      seek_count += distance;
      head = cur_track;
   }

   cout << "Total number of seek operations = " << seek_count << endl;
   cout << "Seek Sequence is" << endl;
   for (int i = 0; i < size; i++) {
      cout << arr[i] << endl;
   }
}

int main()
{
   int arr[size] = { 176, 79, 34, 60, 92, 11, 41, 114 };
   int head = 50;
   FCFS(arr, head);

   return 0;
}
```

OUTPUT:



Request sequence = {176, 79, 34, 60, 92, 11, 41, 114}

Initial head position = 50



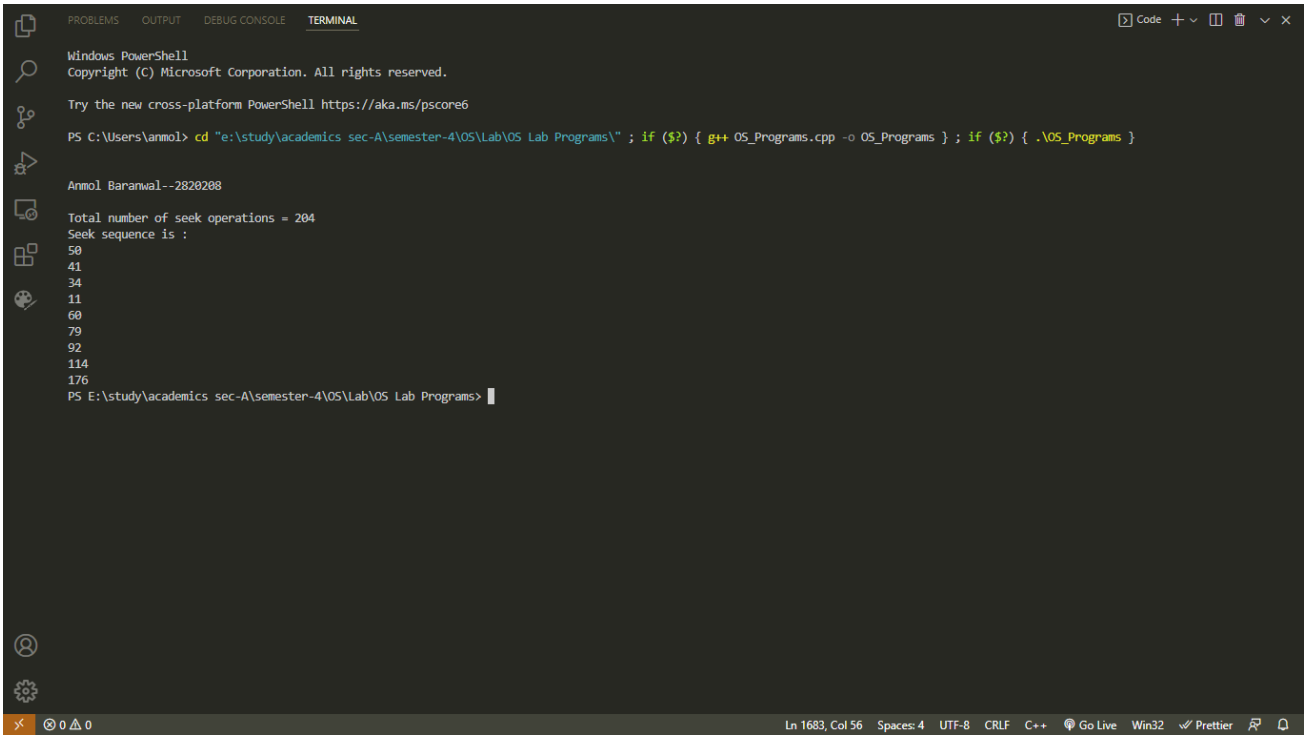
Experiment 13.b.
Write a program to implement SSTF Disk Scheduling Algorithm

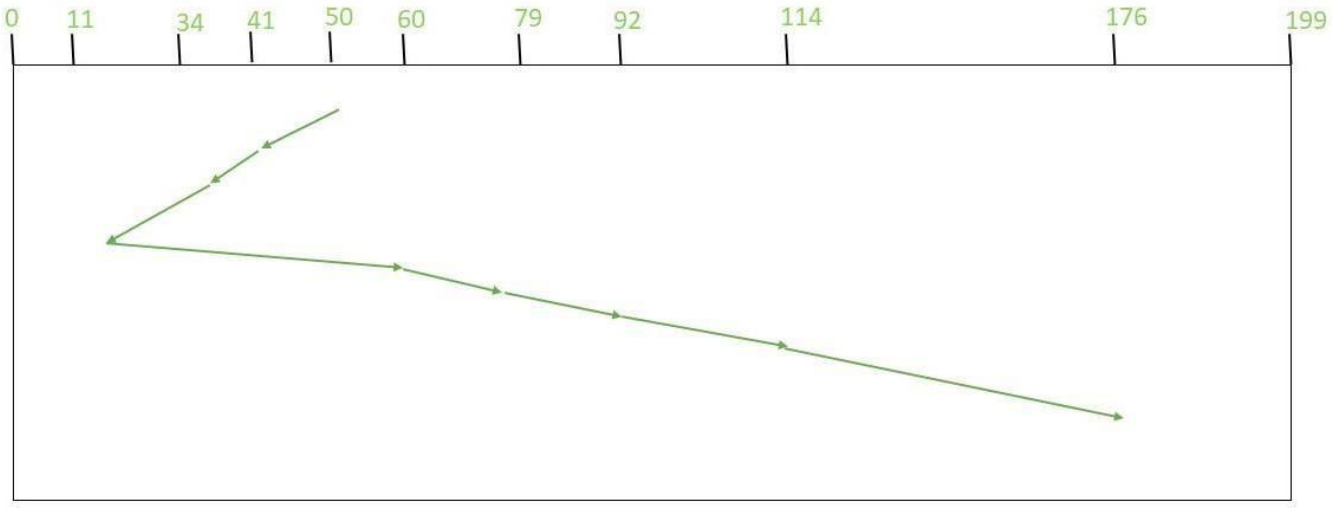
Program:-

```
#include<bits/stdc++.h>
using namespace std;

// Calculates difference of each
// track number with the head position
void calculatedifference(int request[], int head,
    int diff[][2], int n) {
    for (int i = 0; i < n; i++) {
        diff[i][0] = abs(head - request[i]);
    }
}

// Find unaccessed track which is
// at minimum distance from head
int findMIN(int diff[][2], int n)
{
    int index = -1;
    int minimum = 1e9;

    for (int i = 0; i < n; i++) {
        if (!diff[i][1] && minimum > diff[i][0]) {
            minimum = diff[i][0];
            index = i;
        }
    }
    return index;
}

void shortestSeekTimeFirst(int request[],
    int head, int n) {
    if (n == 0) {
        return;
    }
    // Create array of objects of class node
    int diff[n][2] = { { 0, 0 } };
    // Count total number of seek operation
    int seekcount = 0;

    // Stores sequence in which disk access is done
    int seeksequence[n + 1] = {0};
```

```
    for (int i = 0; i < n; i++) {
        seeksequence[i] = head;
        calculatedifference(request, head, diff, n);
        int index = findMIN(diff, n);
        diff[index][1] = 1;
        // Increase the total count
        seekcount += diff[index][0];
        // Accessed track is now new head
        head = request[index];
    }
    seeksequence[n] = head;
    cout << "Total number of seek operations = " <<seekcount << endl;
    cout << "Seek sequence is : " << "\n";

    // Print the sequence
    for (int i = 0; i <= n; i++) {
        cout << seeksequence[i] << "\n";
    }
}

// Driver code
int main()
{
    cout << "\n\nAnmol Baranwal--2820208\n\n";
    int n = 8;
    int proc[n] = { 176, 79, 34, 60, 92, 11, 41, 114 };

    shortestSeekTimeFirst(proc, 50, n);

    return 0;
}
```

OUTPUT:





Request sequence = {176, 79, 34, 60, 92, 11, 41, 114}

Initial head position = 50

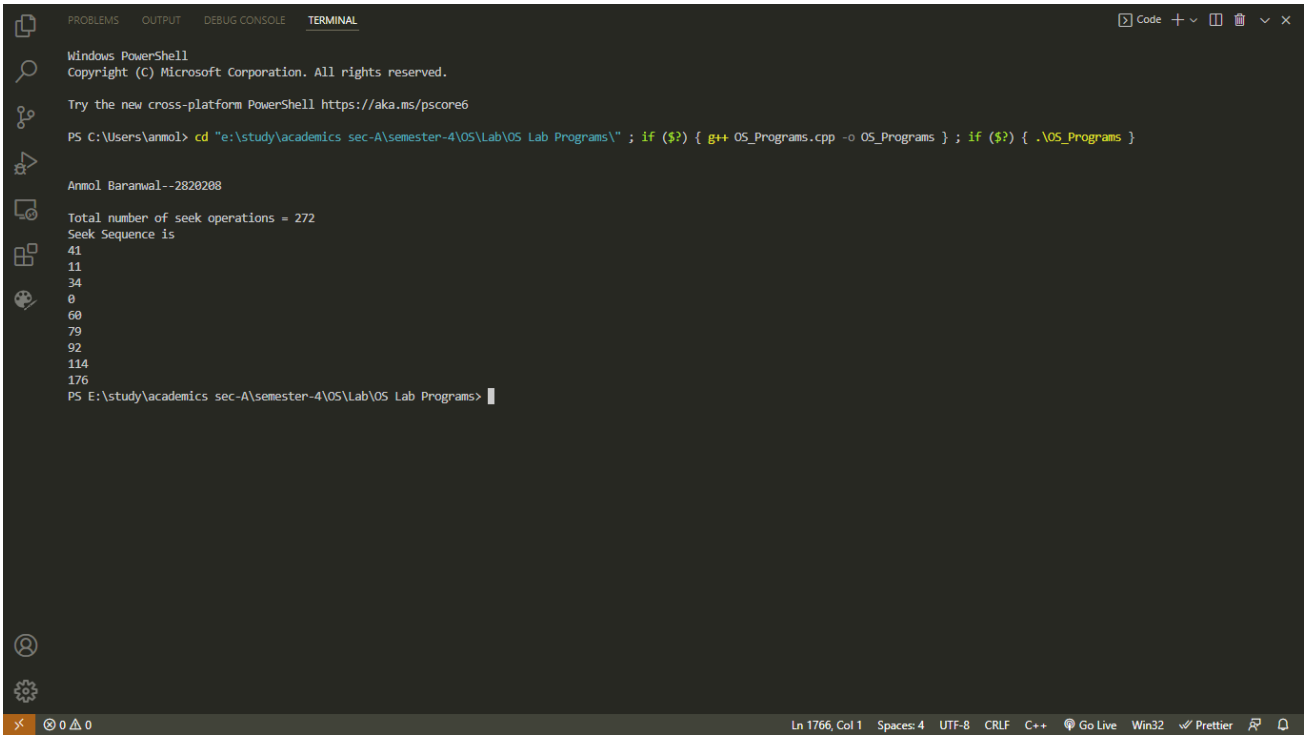
Experiment 13.c.
Write a program to implement SCAN Disk Scheduling

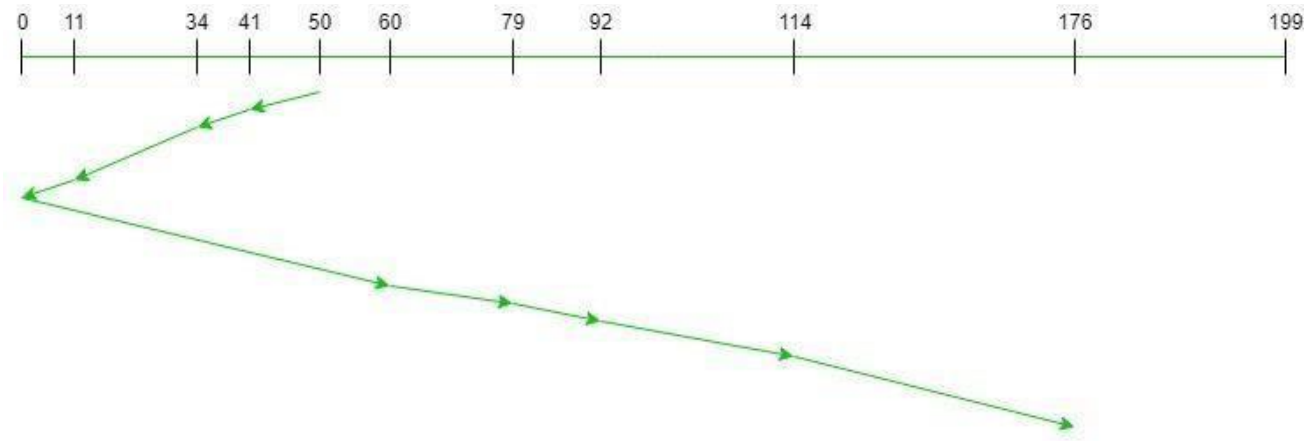
Program:-

```
#include <bits/stdc++.h>
using namespace std;

int size = 8;
int disk_size = 200;

void SCAN(int arr[], int head, string direction) {
    int seek_count = 0;
    int distance, cur_track;
    vector < int > left, right;
    vector < int > seek_sequence;

    // appending end values
    // which has to be visited
    // before reversing the direction
    if (direction == "left")
        left.push_back(0);
    else if (direction == "right") right.push_back(disk_size - 1);

    for (int i = 0; i < size; i++) {
        if (arr[i] < head)
            left.push_back(arr[i]);
        if (arr[i] > head)
            right.push_back(arr[i]);
    }

    // sorting left and right vectors std::sort(left.begin(), left.end());
    std::sort(right.begin(), right.end());

    // run the while loop two times.
    // one by one scanning right
    // and left of the head
    int run = 2;
    while (run--) {
        if (direction == "left") {
            for (int i = left.size() - 1; i >= 0; i--) {
                cur_track = left[i];


                // appending current track to seek sequence
                seek_sequence.push_back(cur_track);
                // calculate absolute distance
                distance = abs(cur_track - head);
```

```
            // increase the total count
            seek_count += distance;
            // accessed track is now the new head
            head = cur_track;
        }
        direction = "right";
    } else if (direction == "right") {
        for (int i = 0; i < right.size(); i++) {
            cur_track = right[i];
            // appending current track to seek sequence
            seek_sequence.push_back(cur_track);
            // calculate absolute distance
            distance = abs(cur_track - head);
            // increase the total count
            seek_count += distance;
            // accessed track is now new head
            head = cur_track;
        }
        direction = "left";
    }
}

cout << "Total number of seek operations = " << seek_count << endl;
cout << "Seek Sequence is" << endl;
for (int i = 0; i < seek_sequence.size(); i++) {
    cout << seek_sequence[i] << endl;
}
}

// Driver code
int main()
{
cout << "\n\nAnmol Baranwal--2820208\n\n";
// request array
int arr[size] = { 176, 79, 34, 60, 92, 11, 41, 114 };
int head = 50;
string direction = "left";
SCAN(arr, head, direction);

return 0;
}
```

OUTPUT:





Request sequence = {176, 79, 34, 60, 92, 11, 41, 114}

Initial head position = 50

Direction = left (We are moving from right to left)

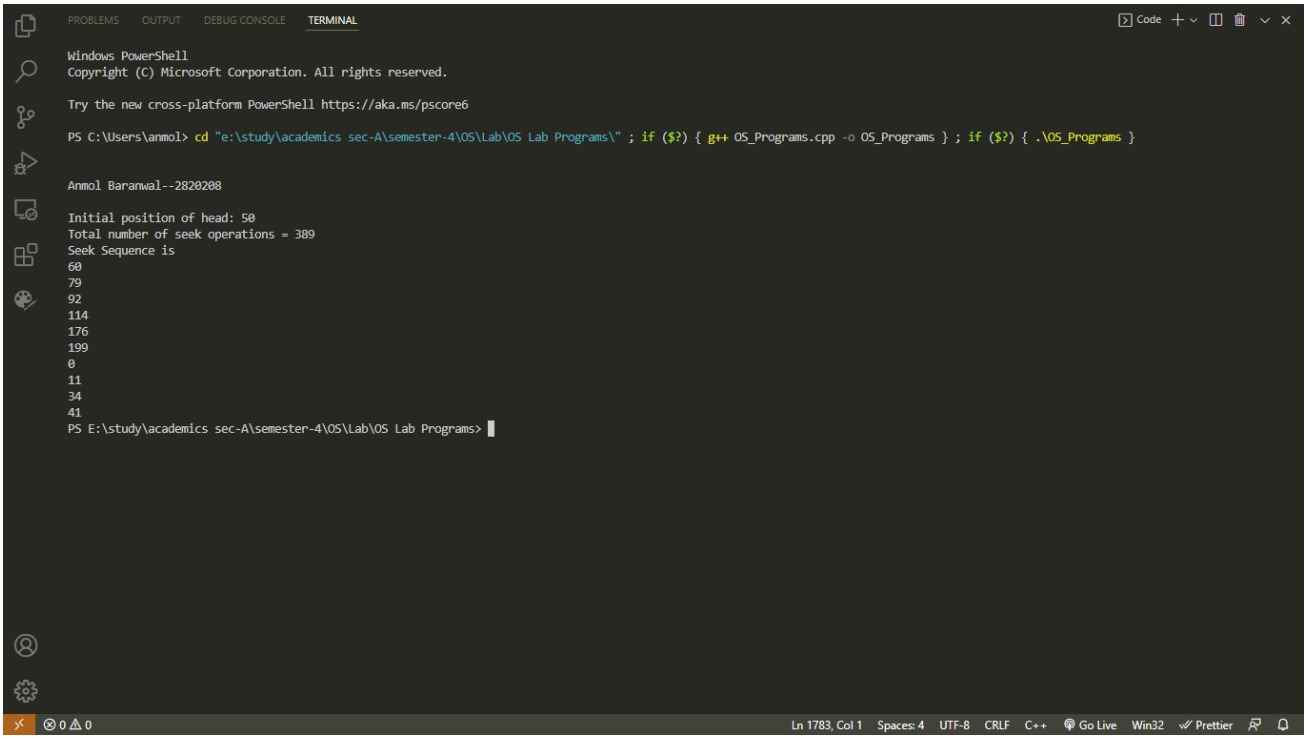
Experiment 13.d.
Write a program to implement C-SCAN Disk Scheduling

Program:-

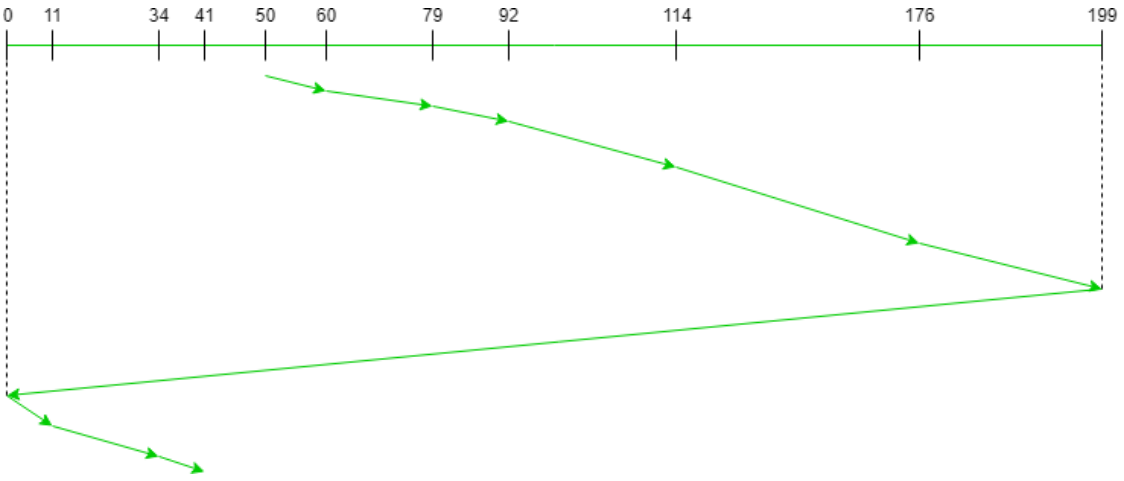
```
#include <bits/stdc++.h>
using namespace std;
int size = 8;
int disk_size = 200;

void CSCAN(int arr[], int head) {
   int seek_count = 0;
   int distance, cur_track;
   vector < int > left, right;
   vector < int > seek_sequence;
   left.push_back(0);
   right.push_back(disk_size - 1);
   for (int i = 0; i < size; i++) {
      if (arr[i] < head) left.push_back(arr[i]);
      if (arr[i] > head) right.push_back(arr[i]);
   }
   std::sort(left.begin(), left.end());
   std::sort(right.begin(), right.end());


   for (int i = 0; i < right.size(); i++) {
      cur_track = right[i];
      seek_sequence.push_back(cur_track);
      distance = abs(cur_track - head);
      seek_count += distance;
      head = cur_track;
   }
   head = 0;
   seek_count += (disk_size - 1);

   for (int i = 0; i < left.size(); i++) {
      cur_track = left[i];
      seek_sequence.push_back(cur_track);
      distance = abs(cur_track - head);

      seek_count += distance;
      head = cur_track;
   }

   cout << "Total number of seek operations = " << seek_count << endl;
   cout << "Seek Sequence is" << endl;
   for (int i = 0; i < seek_sequence.size(); i++) {
```

```
        cout << seek_sequence[i] << endl;
    }
}


int main() {
    cout << "\n\nAnmol Baranwal--2820208\n\n";
    int arr[size] = { 176, 79, 34, 60, 92, 11, 41, 114 };
    int head = 50;
    cout << "Initial position of head: " << head << endl;
    CSCAN(arr, head);
    return 0;
}
```

OUTPUT:




```
Request sequence = {176, 79, 34, 60, 92, 11, 41, 114}
Initial head position = 50
Direction = right(We are moving from left to right)
```



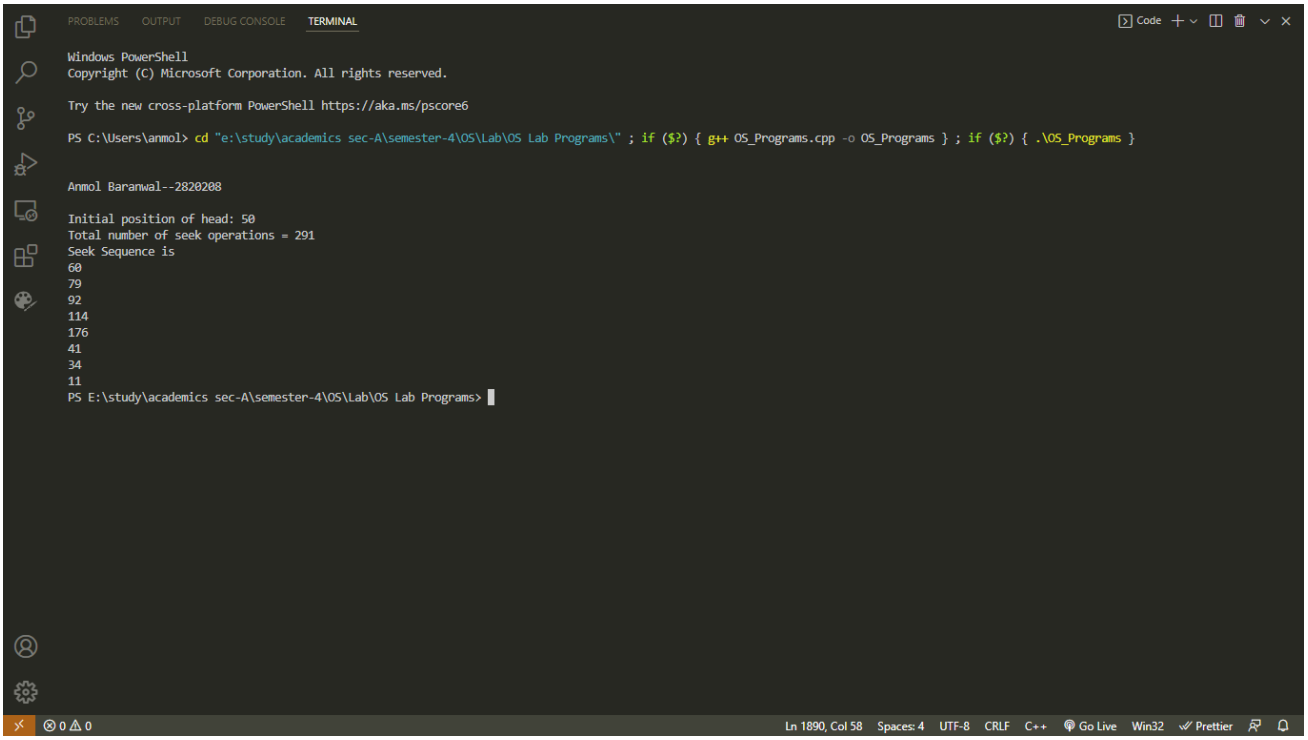
Experiment 13.e.
Write a program to implement LOOK Disk Scheduling

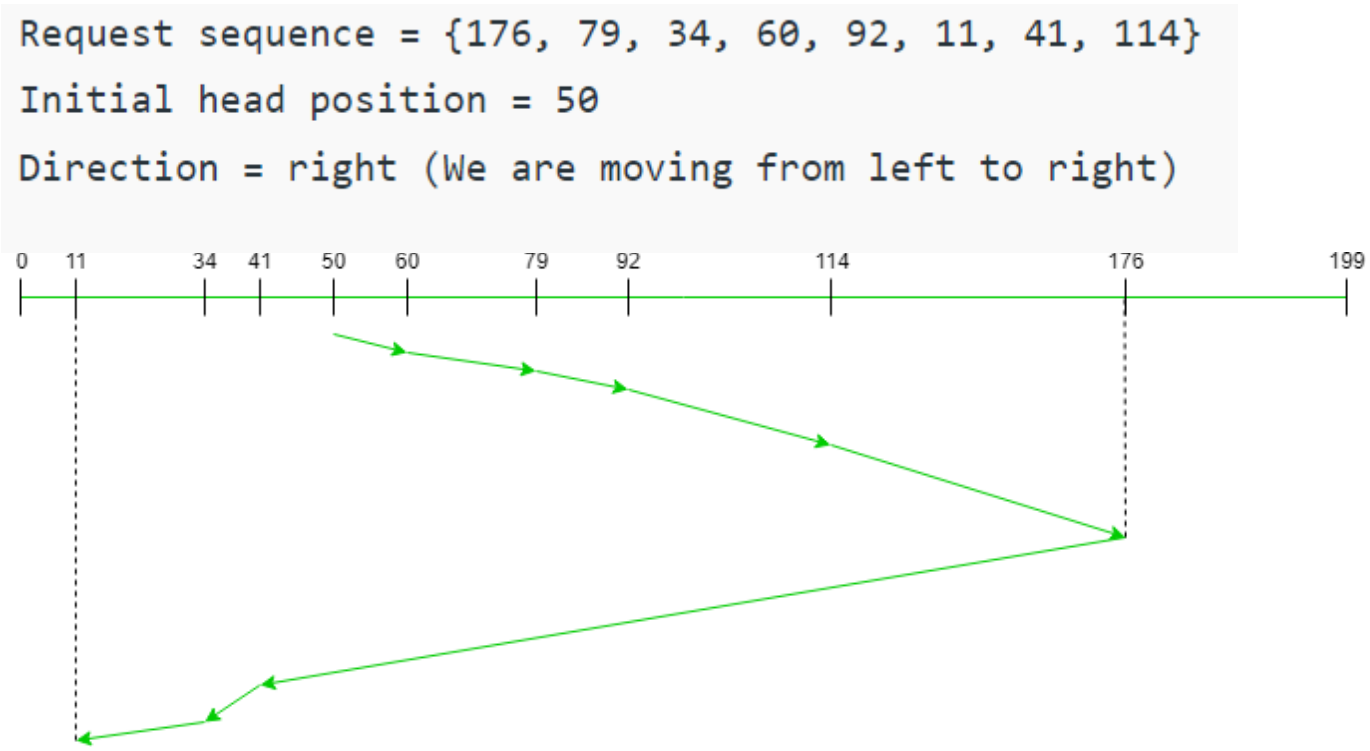
Program:-

```
#include <bits/stdc++.h>
using namespace std;
int size = 8;
int disk_size = 200;

void LOOK(int arr[], int head, string direction) {
    int seek_count = 0;
    int distance, cur_track;
    vector < int > left, right;
    vector < int > seek_sequence;
    for (int i = 0; i < size; i++) {
        if (arr[i] < head)
            left.push_back(arr[i]);
        if (arr[i] > head)
            right.push_back(arr[i]);
    }
    std::sort(left.begin(), left.end());
    std::sort(right.begin(), right.end());

    int run = 2;
    while (run--) {
        if (direction == "left") {
            for (int i = left.size() - 1; i >= 0; i--) {
                cur_track = left[i];
                seek_sequence.push_back(cur_track);
                distance = abs(cur_track - head);
                seek_count += distance;
                head = cur_track;
            }
            direction = "right";
        } else if (direction == "right") {
            for (int i = 0; i < right.size(); i++) {
                cur_track = right[i];
                seek_sequence.push_back(cur_track);

                distance = abs(cur_track - head);
                seek_count += distance;
                head = cur_track;
            }
            direction = "left";
        }
    }
```

```
    cout << "Total number of seek operations = " << seek_count << endl;
    cout << "Seek Sequence is" << endl;

    for (int i = 0; i < seek_sequence.size(); i++) {
        cout << seek_sequence[i] << endl;
    }
}
int main() {
    cout << "\n\nAnmol Baranwal--2820208\n\n";
    int arr[size] = { 176, 79, 34, 60, 92, 11, 41, 114 };
    int head = 50;
    string direction = "right";

    cout << "Initial position of head: " << head << endl;
    LOOK(arr, head, direction);

    return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal--2820208
Initial position of head: 50
Total number of seek operations = 291
Seek Sequence is
60
79
92
114
176
41
34
11
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
Ln 1890, Col 58
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32
Prettier


Request sequence = {176, 79, 34, 60, 92, 11, 41, 114}
Initial head position = 50
Direction = right (We are moving from left to right)
0
11
34
41
50
60
79
92
114
176
199

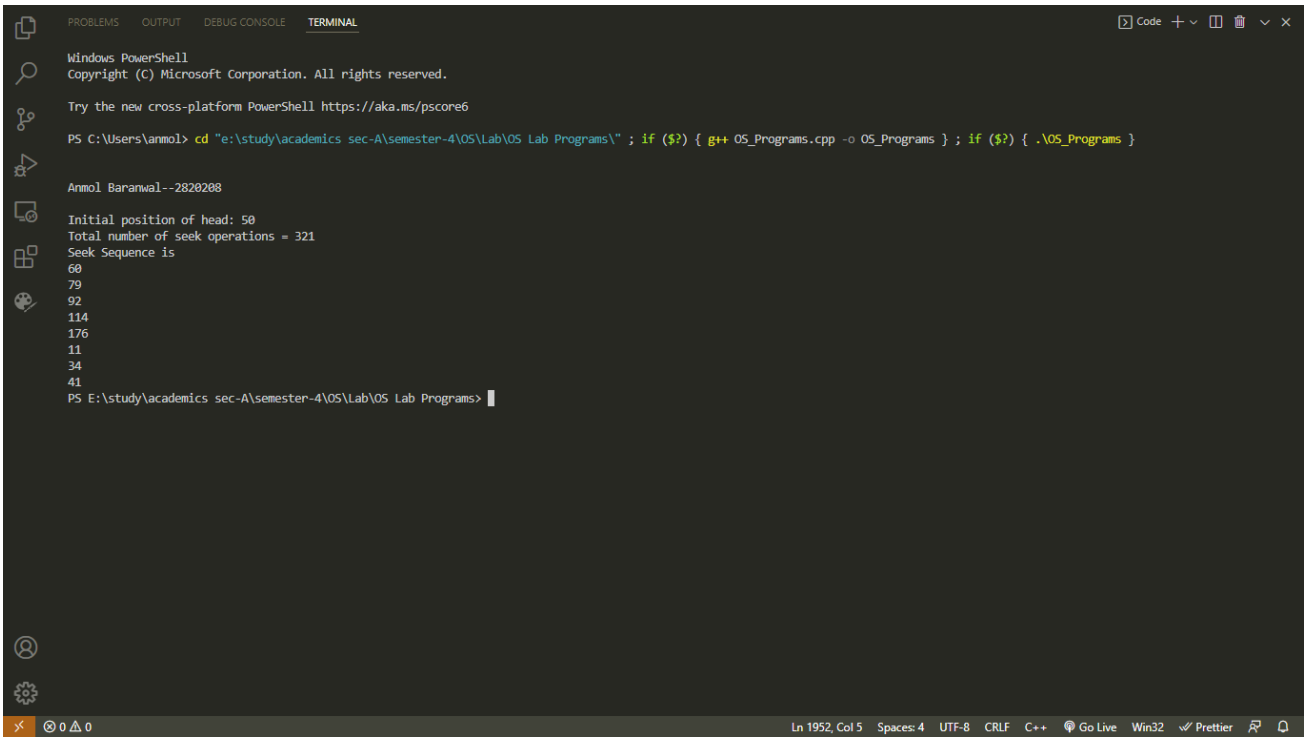
Experiment 13.f.
Write a program to implement C-LOOK Disk Scheduling

Program:-

```
#include <bits/stdc++.h>
using namespace std;
int size = 8;
int disk_size = 200;

void CLOOK(int arr[], int head) {
   int seek_count = 0;
   int distance, cur_track;
   vector < int > left, right;
   vector < int > seek_sequence;

   for (int i = 0; i < size; i++) {
      if (arr[i] < head) left.push_back(arr[i]);
      if (arr[i] > head) right.push_back(arr[i]);
   }
   std::sort(left.begin(), left.end());
   std::sort(right.begin(), right.end());

   for (int i = 0; i < right.size(); i++) {
      cur_track = right[i];

      seek_sequence.push_back(cur_track);
      distance = abs(cur_track - head);
      seek_count += distance;
      head = cur_track;
   }
   seek_count += abs(head - left[0]);
   head = left[0];

   for (int i = 0; i < left.size(); i++) {
      cur_track = left[i];
      seek_sequence.push_back(cur_track);
      distance = abs(cur_track - head);
      seek_count += distance;
      head = cur_track;
   }
   cout << "Total number of seek operations = " << seek_count << endl;
   cout << "Seek Sequence is" << endl;
   for (int i = 0; i < seek_sequence.size(); i++) {
      cout << seek_sequence[i] << endl;
   }
}
```

```
int main() {
    cout << "\n\nAnmol Baranwal--2820208\n\n";
    int arr[size] = { 176, 79, 34, 60, 92, 11, 41, 114 };
    int head = 50;
    cout << "Initial position of head: " << head << endl;
    CLOOK(arr, head);

    return 0;
}
```

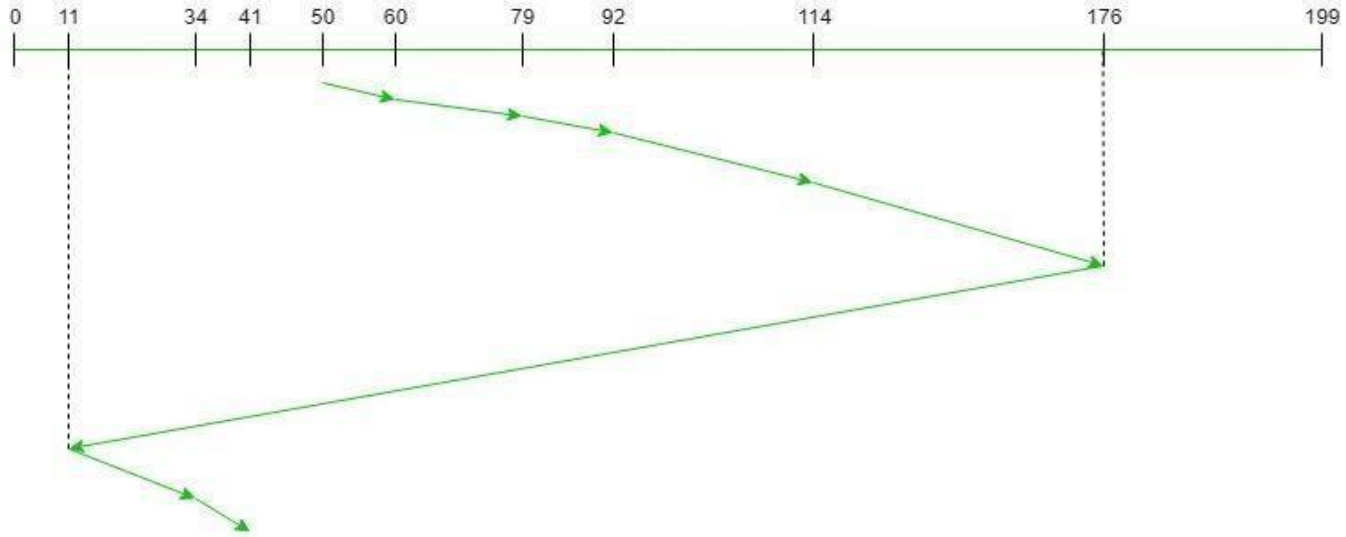
OUTPUT:



Request sequence = {176, 79, 34, 60, 92, 11, 41, 114}

Initial head position = 50

Direction = right (Moving from left to right)



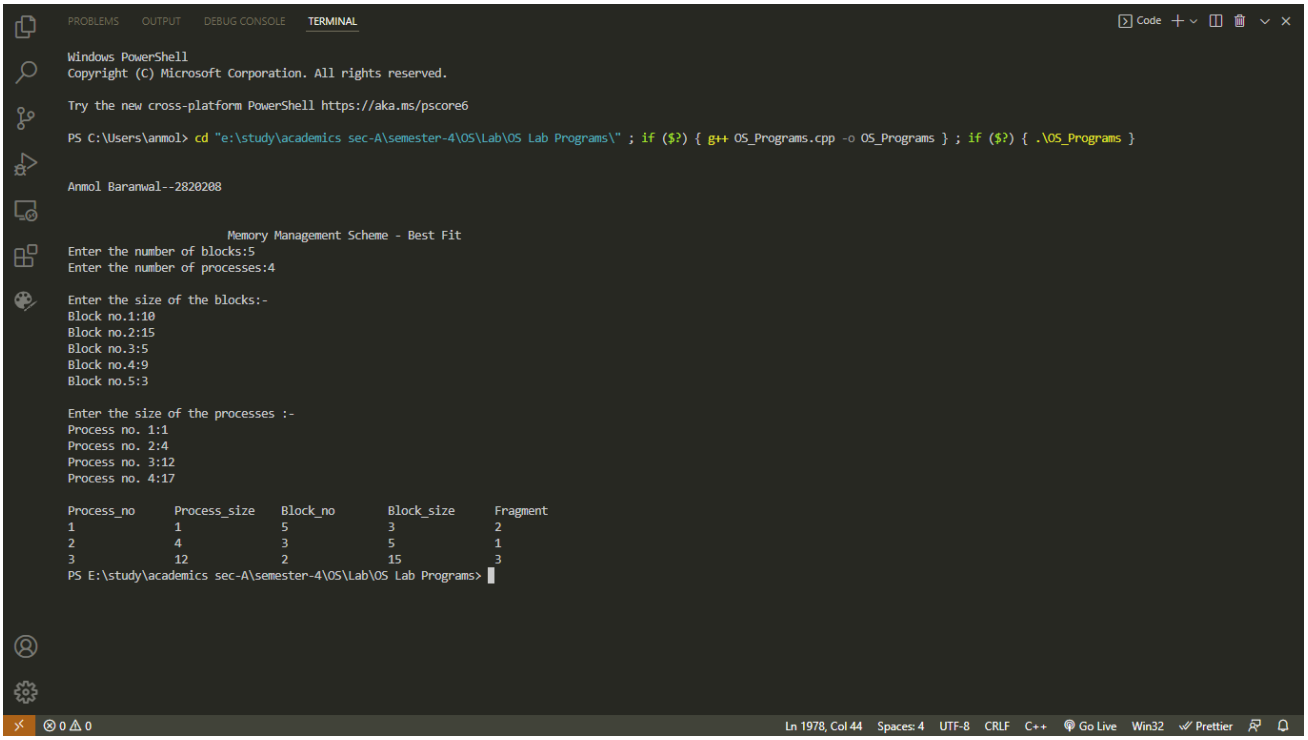
Experiment 14.a.

Write a program to implement First Fit partition allocation method.

Program:-

```
#include<iostream>
using namespace std;

int main() {
   cout << "\n\nAnmol Baranwal--2820208\n\n";
   int fragment[20], b[20], p[20], i, j, nb, np, temp, lowest = 9999;
   static int barray[20], parray[20];
   cout << "\n\t\t\tMemory Management Scheme - Best Fit";
   cout << "\nEnter the number of blocks:";
   cin >> nb;
   cout << "Enter the number of processes:";
   cin >> np;
   cout << "\nEnter the size of the blocks:-\n";
   for (i = 1; i <= nb; i++) {
      cout << "Block no." << i << ":";
      cin >> b[i];
   }
   cout << "\nEnter the size of the processes :-\n";
   for (i = 1; i <= np; i++) {
      cout << "Process no. " << i << ":";
      cin >> p[i];
   }
   for (i = 1; i <= np; i++) {
      for (j = 1; j <= nb; j++) {
         if (barray[j] != 1) {
            temp = b[j] - p[i];
            if (temp >= 0)
               if (lowest > temp) {
                  parray[i] = j;
                  lowest = temp;
               }
         }
      }
      fragment[i] = lowest;
      barray[parray[i]] = 1;
      lowest = 10000;
   }
   cout << "\nProcess_no\tProcess_size\tBlock_no\tBlock_size\tFragment";
   for (i = 1; i <= np && parray[i] != 0; i++)
      cout << "\n" << i << "\t\t" << p[i] << "\t\t" << parray[i] << "\t\t" << b[parray[i]] << "\t\t" <<
fragment[i];
   return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal--2820208
Memory Management Scheme - Best Fit
Enter the number of blocks:5
Enter the number of processes:4
Enter the size of the blocks:-
Block no.1:10
Block no.2:15
Block no.3:5
Block no.4:9
Block no.5:3
Enter the size of the processes :-
Process no. 1:1
Process no. 2:4
Process no. 3:12
Process no. 4:17
Process_no Process_size Block_no Block_size Fragment
1 1 5 3 2
2 4 3 5 1
3 12 2 15 3
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 1978, Col 44
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32
Prettier

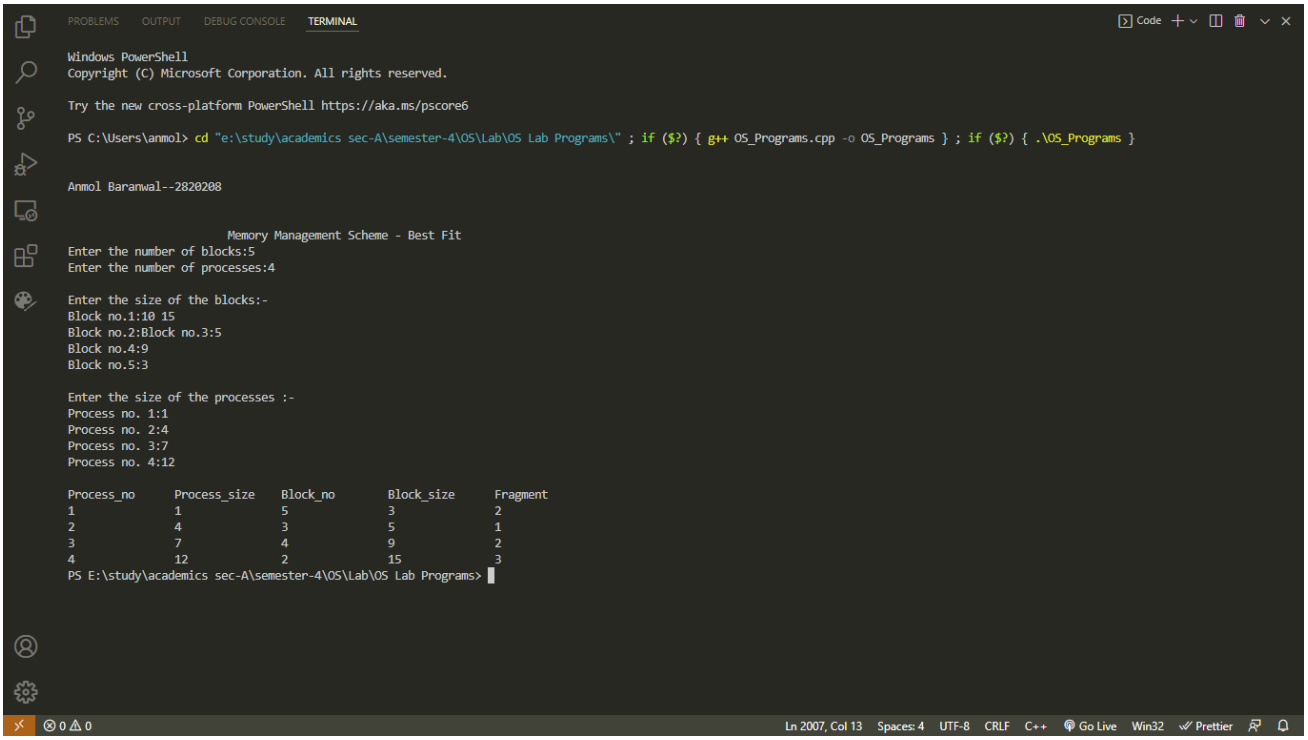
Experiment 14.b.

Write a program to implement Best Fit partition allocation method.

Program:-

```
#include<iostream>
using namespace std;

int main() {
   cout << "\n\nAnmol Baranwal--2820208\n\n";
   int fragment[20], b[20], p[20], i, j, nb, np, temp, lowest = 9999;
   static int barray[20], parray[20];
   cout << "\n\t\t\tMemory Management Scheme - Best Fit";
   cout << "\nEnter the number of blocks:";
   cin >> nb;
   cout << "Enter the number of processes:";
   cin >> np;
   cout << "\nEnter the size of the blocks:-\n";
   for (i = 1; i <= nb; i++) {
      cout << "Block no." << i << ":";
      cin >> b[i];
   }
   cout << "\nEnter the size of the processes :-\n";
   for (i = 1; i <= np; i++) {
      cout << "Process no. " << i << ":";
      cin >> p[i];
   }
   for (i = 1; i <= np; i++) {
      for (j = 1; j <= nb; j++) {
         if (barray[j] != 1) {
            temp = b[j] - p[i];
            if (temp >= 0)
               if (lowest > temp) {
                  parray[i] = j;
                  lowest = temp;
               }
         }
      }
      fragment[i] = lowest;
      barray[parray[i]] = 1;
      lowest = 10000;
   }
   cout << "\nProcess_no\tProcess_size\tBlock_no\tBlock_size\tFragment";
   for (i = 1; i <= np && parray[i] != 0; i++)
      cout << "\n" << i << "\t\t" << p[i] << "\t\t" << parray[i] << "\t\t" << b[parray[i]] << "\t\t" <<
fragment[i];
   return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal--2820208
Memory Management Scheme - Best Fit
Enter the number of blocks:5
Enter the number of processes:4
Enter the size of the blocks:-
Block no.1:10 15
Block no.2:Block no.3:5
Block no.4:9
Block no.5:3
Enter the size of the processes :-
Process no. 1:1
Process no. 2:4
Process no. 3:7
Process no. 4:12
Process_no Process_size Block_no Block_size Fragment
1 1 5 3 2
2 4 3 5 1
3 7 4 9 2
4 12 2 15 3
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 2007, Col 13
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32
Prettier

Experiment 14.c.

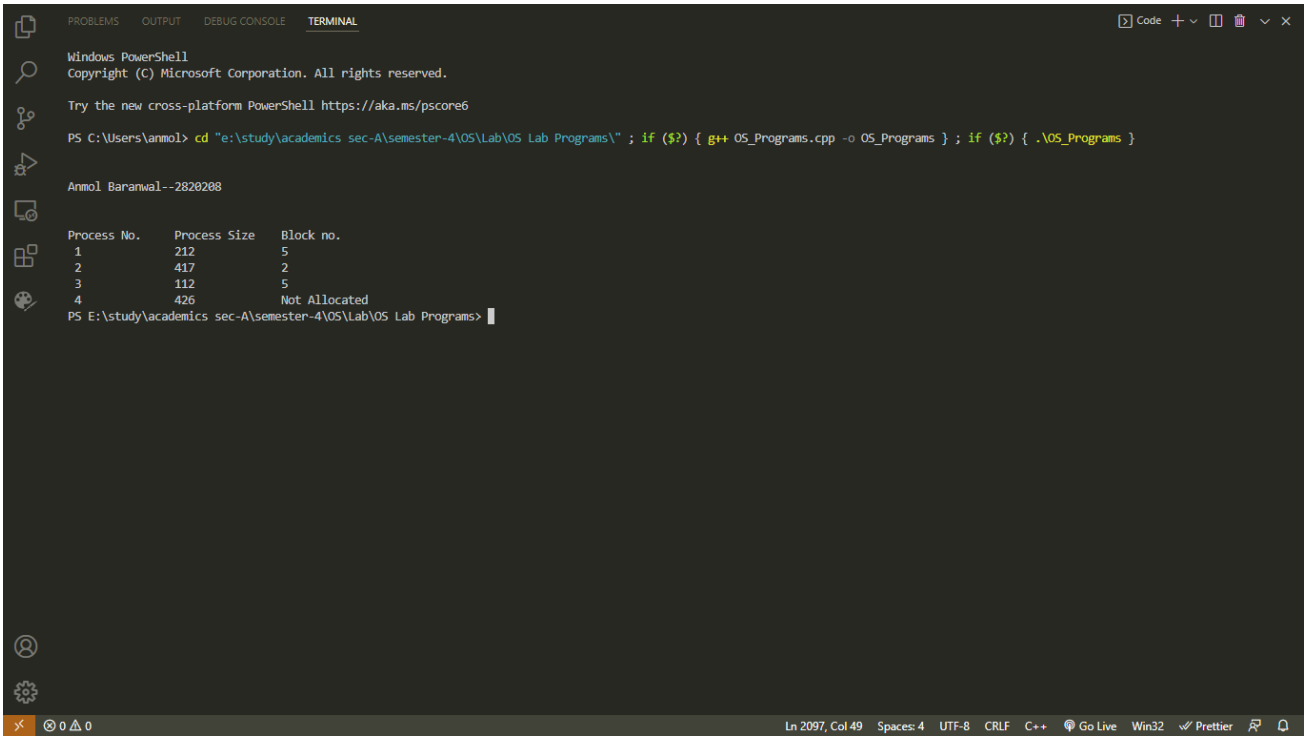
Write a program to implement Worst Fit partition allocation method.

Program:-

```
#include<bits/stdc++.h>
using namespace std;

// Function to allocate memory to blocks as per worst fit algorithm
void worstFit(int blockSize[], int m, int processSize[], int n) {
    // Stores block id of the block allocated to a process
    int allocation[n];
    // Initially no block is assigned to any process
    memset(allocation, -1, sizeof(allocation));

    // pick each process and find suitable blocks
    // according to its size ad assign to it
    for (int i=0; i<n; i++)
    {
        // Find the best fit block for current process
        int wstIdx = -1;
        for (int j = 0; j < m; j++) {
            if (blockSize[j] >= processSize[i]) {
                if (wstIdx == -1)
                    wstIdx = j;
                else if (blockSize[wstIdx] < blockSize[j]) wstIdx = j;
            }
        }
        // If we could find a block for current process
        if (wstIdx != -1)
        {
            // allocate block j to p[i] process
            allocation[i] = wstIdx;
            // Reduce available memory in this block.
            blockSize[wstIdx] -= processSize[i];
        }
    }
    cout << "\nProcess No.\tProcess Size\tBlock no.\n";
    for (int i = 0; i < n; i++) {
        cout << " " << i + 1 << "\t\t" << processSize[i] << "\t\t";
        if (allocation[i] != -1)
            cout << allocation[i] + 1;
        else
            cout << "Not Allocated";
        cout << endl;
    }
}
```

```
// Driver code
int main()
{
    cout << "\n\nAnmol Baranwal--2820208\n\n";
    int blockSize[] = {100, 500, 200, 300, 600};
    int processSize[] = {212, 417, 112, 426};
    int m = sizeof(blockSize) / sizeof(blockSize[0]);
    int n = sizeof(processSize) / sizeof(processSize[0]);

    worstFit(blockSize, m, processSize, n);

    return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal--2820208
Process No. Process Size Block no.
1 212 5
2 417 2
3 112 5
4 426 Not Allocated
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 2097, Col 49
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32
Prettier

Experiment 14.d.

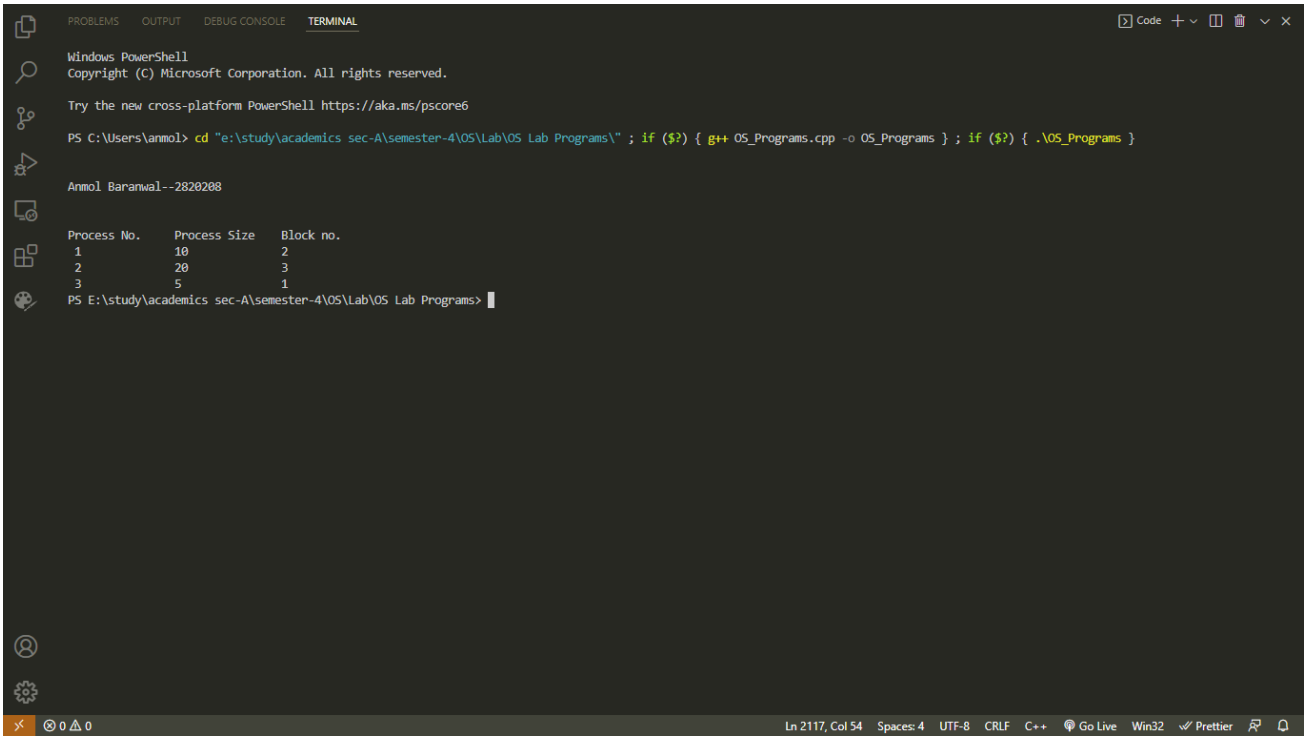
Write a program to implement Next Fit partition allocation method.

Program:-

```
#include <bits/stdc++.h>
using namespace std;

// Function to allocate memory to blocks as per Next fit algorithm
void NextFit(int blockSize[], int m, int processSize[], int n) {
    // Stores block id of the block allocated to a process
    int allocation[n], j = 0;
    // Initially no block is assigned to any process
    memset(allocation, -1, sizeof(allocation));
    // pick each process and find suitable blocks
    // according to its size ad assign to it
    for (int i = 0; i < n; i++) {

        // Do not start from beginning
        while (j < m) {
            if (blockSize[j] >= processSize[i]) {
                // allocate block j to p[i] process
                allocation[i] = j;

                // Reduce available memory in this block.
                blockSize[j] -= processSize[i];
                break;
            }
            // mod m will help in traversing the blocks from
            // starting block after we reach the end.
            j = (j + 1) % m;
        }
    }

    cout << "\nProcess No.\tProcess Size\tBlock no.\n";
    for (int i = 0; i < n; i++) {
        cout << " " << i + 1 << "\t\t" << processSize[i] <<
            "\t\t";
        if (allocation[i] != -1)
            cout << allocation[i] + 1;
        else
            cout << "Not Allocated";
        cout << endl;
    }
}

// Driver program
```

```
int main() {
    cout << "\n\nAnmol Baranwal--2820208\n\n";
    int blockSize[] = { 5, 10, 20 };
    int processSize[] = { 10, 20, 5 };
    int m = sizeof(blockSize) / sizeof(blockSize[0]);
    int n = sizeof(processSize) / sizeof(processSize[0]);

    NextFit(blockSize, m, processSize, n);

    return 0;
}
```

OUTPUT:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\anmol> cd "e:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs\" ; if ($?) { g++ OS_Programs.cpp -o OS_Programs } ; if ($?) { .\OS_Programs }
Anmol Baranwal--2820208
Process No. Process Size Block no.
1 10 2
2 20 3
3 5 1
PS E:\study\academics sec-A\semester-4\OS\Lab\OS Lab Programs>
0 0
Ln 2117, Col 54
Spaces: 4
UTF-8
CRLF
C++
Go Live
Win32
Prettier