DEPARTMENT OF ACADEMIC AFFAIRS
Discover. Learn. Empower.

NAAC GRADE A+ ACCREDITED UNIVERSITY

Experiment 10

Student Name: V.Sri Surya Prakash Reddy **UID:** 21BCS9133
Branch: CSE AIML **Section/Group: 12**-A
Semester: 3rd **Date of Performance: 9** November
Subject Name: Operating Systems **Subject Code:** 21CSH - 242

Aim/Overview of the practical:

1. **Simulation of Banker's Algorithm.**

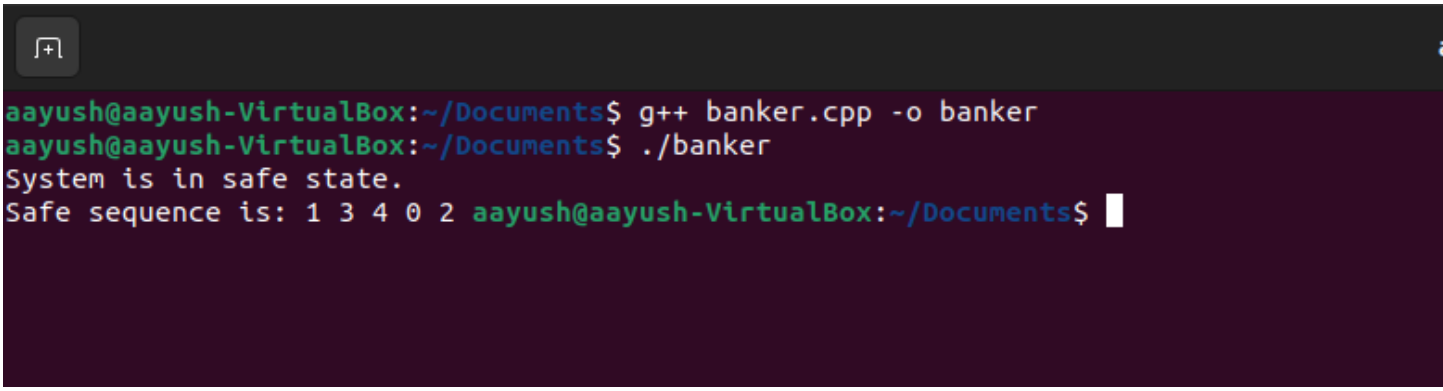
- **Code:**

```
#include<iostream>
using namespace std;
const int P = 5;
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()
{
	int processes[] = {0, 1, 2, 3, 4};
	int avail[] = {3, 3, 2};
	int maxm[][R] = {{7, 5, 3},
				{3, 2, 2},
				{9, 0, 2},
				{2, 2, 2},
				{4, 3, 3}};
	int allot[][R] = {{0, 1, 0},
				{2, 0, 0},
				{3, 0, 2},
				{2, 1, 1},
				{0, 0, 2}};
	isSafe(processes, avail, maxm, allot);
	return 0;
}
```

- **Screenshots:**



- **Result/Output/Wri ting Summary:**

  The commands on the Nano editor worked perfectly fine

Learning outcomes (What I have learnt):

- Learnt how to use Nano editor.
- Learnt some basic commands in Linux terminal • Learnt some basic commands used in Nano editor.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks