

OPERITING SYSTEMS LAB 2

BANKER'S ALGORITHM



Team :-

- | | |
|---|----------|
| - Adel Mohamed Mahmoud Hashem | 18010880 |
| - Mostafa Ahmed Abd EL-Salam EL-Hossary | 18011775 |
| - Mahmoud Ahmed Abd EL-Salam Mohamed | 18011654 |

CODE :-

- THE MAIN FUNCTION HERE IS "BANKER" IT STANDS FOR BANKER'S ALGORITHM :

```
85 void Banker(int C[5][3],int M[5][3],int A[1][3],int need[5][3])
86 {
87     Queue p,seq;
88     CreateQueue(&p,5);
89     CreateQueue(&seq,5);
90     for(int i = 0;i<5;i++)
91         QueueAdd(&p , i);
92
93     while(!IsEmpty(&p))
94     {
95         int found = 0,p_size = p.size;
96         for(int i = 0 ; i < p_size;i++)
97         {
98             int process = QueueOut(&p);
99             // Can Give Resources To That process
100             if(need[process][0] <= A[0][0] && need[process][1] <= A[0][1] && need[process][2] <= A[0][2])
101             {
102                 // Add Take The Resources Back From The Process
103                 for (int j = 0; j < 3; ++j) {
104                     A[0][j] = A[0][j] + C[process][j];
105                 }
106                 found = 1;
107                 QueueAdd(&seq,process); //Add the process to Sequence Queue
108             }else //Not Enough Resources
109                 QueueAdd(&p,process); //Append it Back to the Queue
110
111         }
112         if (!found)
113         {
114             printf("\n Unsafe State\n");
115             return;
116         }
117     }
118
119     //Print the Sequence
120     printf("\n Safe State\n");
121     printf("\n possible execution sequence : ");
122     QueuePrint(seq);
123
124 }
125 }
```

- TO GET THE DATA FROM THE FILE WE NEEDED TO USE SOME FUNCTIONS :

1- Loadfile : to load the file we got from the user.

```
50 void LoadFile(char *path)
51 {
52     getcwd(path,sizeof(path)); //go to the current path
53     printf("Enter The File Name ==> ");
54     char filename[50];
55     scanf("%s",filename); // get the file name from the user
56     strcat(path,filename);
57 }
```

2- *Getdata* : to get specific data from the file.

```
59 void GetData(int n,int c[][3])
60 {
61     int temp = 0,i=0;
62
63     while (i<n)
64     {
65         while(temp < 3)
66             fscanf(f, "%d", &c[i][temp++]);
67         temp = 0;
68         i++;
69     }
70 }
71 }
```

3- *subtract_matrices* : to separate each matrix (Cp,Mp,A).

```
73 void subtract_matrices(int rows,int col,int out[][col],int mat1[][col],int mat2[][col])
74 {
75     // subtract two matrices
76     for (int i = 0; i < rows; ++i)
77     {
78         for (int j = 0; j < col; ++j) {
79             out[i][j] = mat1[i][j] - mat2[i][j];
80         }
81     }
82 }
```


- WE USED QUEUE DATA STRACT. TO HANDLE THE PROCESSES :

```
6 //Build Queue
7 typedef struct {
8     int front;
9     int rear;
10    int size;
11    int max;
12    int *data;
13 }Queue;
14
15 void CreateQueue(Queue* q ,int n);
16 int IsEmpty(Queue* q); //check if queue is empty
17 void QueueAdd(Queue* q , int ndata);
18 int QueueOut(Queue* q);
19 void QueuePrint (Queue q);
20 }
```

SAMPLE RUNS :-

- WE TRIED A LOT OF EXAMPLES ,HERE'S TWO OF THEM :-

1- *b.txt* file

 b.txt - Notepad

File Edit Format View

0 1 0

2 0 0

3 0 2

2 1 1

0 0 2

7 5 3

3 2 2

9 0 2

2 2 2

4 3 3

3 3 2

The run :-


```
C:\Users\moamen\Desktop\Banker-s-Algorithm-main\banker.exe
Enter The File Name ==> b.txt

Safe State

possible execution sequence : P1,P3,P4,P0,P2

Process returned 0 (0x0)   execution time : 6.801 s
Press any key to continue.
```

2- SnapShot.txt file

 SnapShot.txt -

File Edit Format

1 0 0

6 1 2

2 1 1

0 0 2

0 0 0

3 2 2

6 1 3

3 1 4

4 2 2

5 2 5

0 1 1

The run :-

```
C:\Users\moamen\Desktop\Banker-s-Algorithm-main\banker.exe
Enter The File Name ==> SnapShot.txt

Safe State

possible execution sequence : P1,P2,P3,P4,P0

Process returned 0 (0x0)   execution time : 21.904 s
Press any key to continue.
_
```