LAB 10

Course: CT-353-Operating Systems

Department: BCIT (Specialisation in Data Science)

Instructor's Name: Muhammed Abdullah Siddiqui

Student Name: Maryam Ashraff (DT-22050)

CODE:

```cpp
[*] OS LAB 10.cpp
 1    #include <iostream>
 2    using namespace std;
 3
 4    int main() {
 5        int ms, ps, nop, np, rempages;
 6        int s[10], fno[10][20];
 7        int i, j, x, y, pa, offset;
 8
 9        // Input total memory size
10        cout << "\nEnter the memory size -- ";
11        cin >> ms;
12
13        // Input page size
14        cout << "\nEnter the page size -- ";
15        cin >> ps;
16
17        nop = ms / ps;   // Number of pages in memory
18        cout << "\nThe number of pages available in memory are -- " << nop;
19
20        // Input number of processes
21        cout << "\nEnter number of processes -- ";
22        cin >> np;
23
24        rempages = nop;
25
26        // Input for each process
27        for (i = 1; i <= np; i++) {
28            cout << "\nEnter number of pages required for p[" << i << "] -- ";
29            cin >> s[i];
30
31            if (s[i] > rempages) {
32                cout << "\nMemory is Full";
33                break;
34            }
35
36            rempages -= s[i];
37            cout << "\nEnter page table for p[" << i << "] ---\n";
38
39            for (j = 0; j < s[i]; j++) {
40                cout << "Page " << j << " ? Frame: ";
41                cin >> fno[i][j];
42            }
43        }
44
45        // Logical to Physical Address translation
46        cout << "\nEnter Logical Address to find Physical Address";
47        cout << "\nEnter process no., page number and offset -- ";
48        cin >> x >> y >> offset;
49
50        if (x > np || y >= s[x] || offset >= ps) {
51            cout << "\nInvalid Process or Page Number or Offset";
52        } else {
53            pa = fno[x][y] * ps + offset;
54            cout << "\nThe Physical Address is -- " << pa;
55        }
56
57        return 0;
58    }
59
```

OUTPUT:

```
C:\Users\marya\Downloads\O

Enter the memory size -- 1000

Enter the page size -- 100

The number of pages available in memory are -- 10
Enter number of processes -- 2

Enter number of pages required for p[1] -- 3

Enter page table for p[1] ---
Page 0 ? Frame: 5 6 7
Page 1 ? Frame: Page 2 ? Frame:
Enter number of pages required for p[2] -- 4

Enter page table for p[2] ---
Page 0 ? Frame: 1 2 3 4
Page 1 ? Frame: Page 2 ? Frame: Page 3 ? Frame:
Enter Logical Address to find Physical Address
Enter process no., page number and offset -- 1 2 50

The Physical Address is -- 750
--------------------------------
Process exited after 41.58 seconds with return value 0
Press any key to continue . . .
```