LAB 10

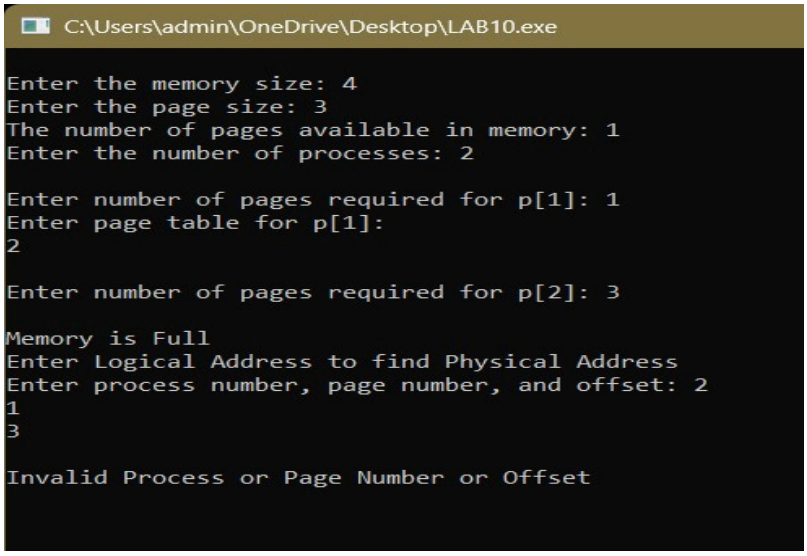
Implement the above code and paste the screen shot of the output.

CODE:

```
#include <stdio.h>
#include <conio.h>

int main() {
    int ms, ps, nop, np, rempages, i, j, x, y, pa, offset;
    int s[10], fno[10][20];



    printf("\nEnter the memory size: ");
    scanf("%d", &ms);

    printf("Enter the page size: ");
    scanf("%d", &ps);

    nop = ms / ps;
    printf("The number of pages available in memory: %d", nop);

    printf("\nEnter the number of processes: ");
    scanf("%d", &np);

    rempages = nop;

    for (i = 1; i <= np; i++) {
        printf("\nEnter number of pages required for p[%d]: ", i);
        scanf("%d", &s[i]);
```

```
        if (s[i] > rempages) {
            printf("\nMemory is Full");
            break;
        }

        rempages -= s[i];

        printf("Enter page table for p[%d]:\n", i);
        for (j = 0; j < s[i]; j++) {
            scanf("%d", &fno[i][j]);
        }
    }

    printf("\nEnter Logical Address to find Physical Address");
    printf("\nEnter process number, page number, and offset: ");
    scanf("%d %d %d", &x, &y, &offset);

    if (x > np || y >= s[x] || offset >= ps) {
        printf("\nInvalid Process or Page Number or Offset");
    } else {
        pa = fno[x][y] * ps + offset;
        printf("The Physical Address is: %d", pa);
    }

    getch();
    return 0;
}
```

OUTPUT:


C:\Users\admin\OneDrive\Desktop\LAB10.exe
Enter the memory size: 4
Enter the page size: 3
The number of pages available in memory: 1
Enter the number of processes: 2
Enter number of pages required for p[1]: 1
Enter page table for p[1]:
2
Enter number of pages required for p[2]: 3
Memory is Full
Enter Logical Address to find Physical Address
Enter process number, page number, and offset: 2
1
3
Invalid Process or Page Number or Offset