

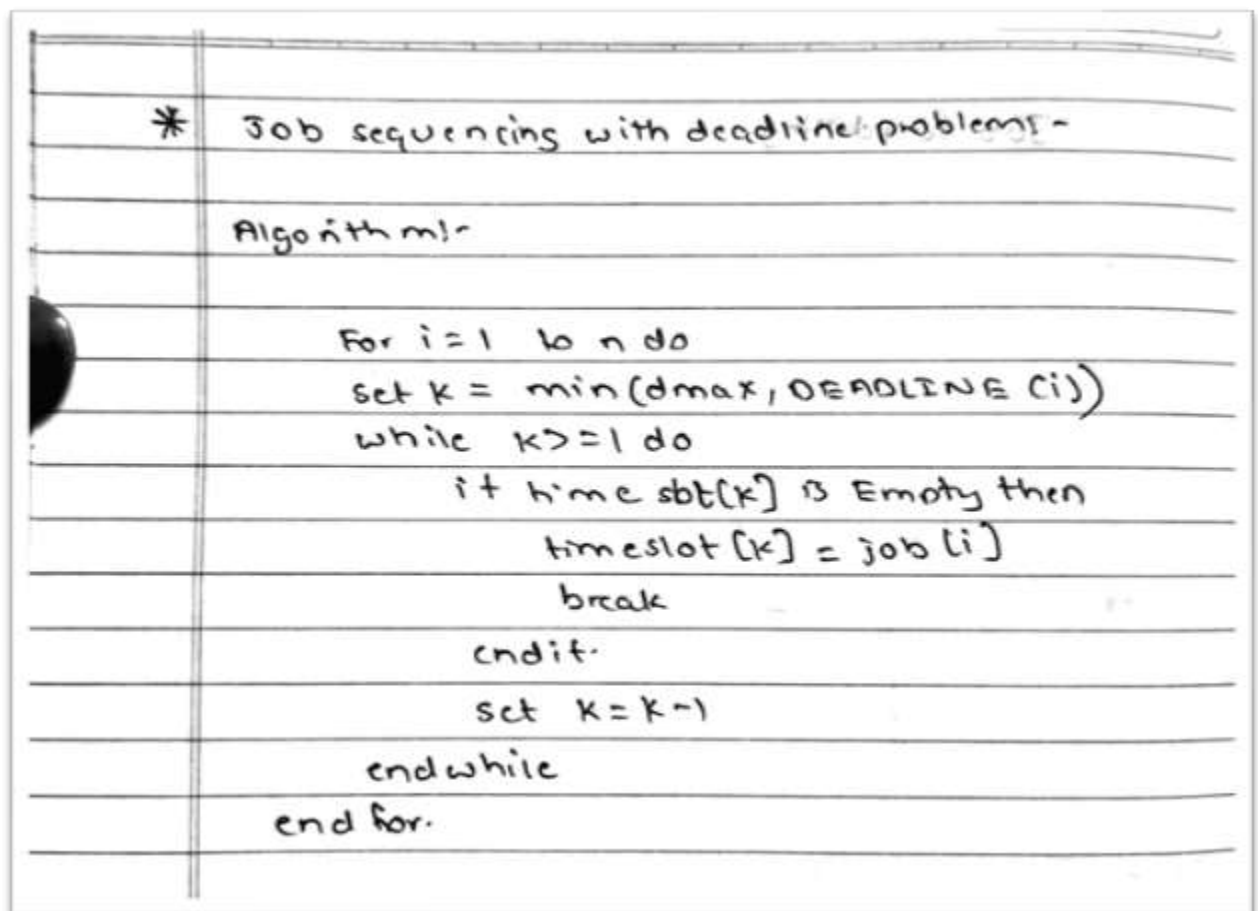
# Experiment No. 6

**Aim:** Write a program to implement Job sequencing with deadlines problem.

## Theory:

Given a set of tasks with deadlines and total profit earned on completing task, find maximum point earned by execution the tasks within the specified deadlines. Assume that a task takes one unit of time to execute, and it can't execute beyond its deadline. Also, only single task will be executed at a time.

## Algorithm:



## Code:

```
using namespace std;
#include <iostream>
#include <vector>
#include <algorithm>
#include <numeric>

int main(){
    int n;

    int d,p;
    int max;

    cout << "Enter the number jobs ";
    cin >> n;
    vector<pair<int,int>> a(n);
    vector<int> b(n);
    cout << "Profit Deadline" << endl;
    for(int i=0;i<n;i++){
        cin >> p;
        cin >> d;
        a.push_back(make_pair(p,d));
    }

    sort(a.rbegin(),a.rend());

    for(int i=0;i<n;i++){
        if(b[a[i].second-1] == 0){
            b[a[i].second-1]=a[i].first;
        }else {
            for(int j=0;j<i;j++){
                if(b[j]== 0 && j != a[i].second-1 && j < a[i].second-1){
                    b[j]=a[i].first;
                }
            }
        }
    }
    b.erase(remove(begin(b), end(b), 0), end(b));
    max = accumulate(b.begin(), b.end(), 0);
    cout << "Sort Job Schedule " << endl;
    for(int x : b){
        cout << x << " ";
    }
    cout << endl;
    cout << "Max Profit " << max << endl;
    return 0;
}
```

## Output:

```
Enter the number jobs 5
Profit  Deadline
100 2
19 1
27 2
25 1
15 3
Sort Job Schedule
27 100 15
Max Profit 142
PS C:\Users\chinm\Downloads>
```

## Time Complexity:

Job sequencing problems has the time complexity of  $O(n^2)$ .

## Conclusion:

Hence, we studied the implementation of Job sequencing with deadlines problem.