```
#include <iostream>
using namespace std;

void jobScheduling(int n, int jobs[], int deadline[], int profit[]) {
    // Find maximum deadline
    int maxDeadline = 0;
    for (int i = 0; i < n; i++) {
        if (deadline[i] > maxDeadline)
            maxDeadline = deadline[i];
    }

    // Initialize time slots and result array
    int slot[maxDeadline + 1]; // 1-based index
    for (int i = 1; i <= maxDeadline; i++)
        slot[i] = -1; // -1 means free

    // Sort jobs based on profit (descending) using simple selection sort
    for (int i = 0; i < n - 1; i++) {
        for (int j = i + 1; j < n; j++) {
            if (profit[i] < profit[j]) {
                swap(profit[i], profit[j]);
                swap(deadline[i], deadline[j]);
                swap(jobs[i], jobs[j]);
            }
        }
    }
    int totalProfit = 0;
    cout << "Selected jobs: ";
    for (int i = 0; i < n; i++) {
        // Find a free slot for this job before or on its deadline
        for (int j = deadline[i]; j > 0; j--) {
            if (slot[j] == -1) {
                slot[j] = jobs[i];
                totalProfit += profit[i];
                cout << "Job" << jobs[i] << " ";
                break;
            }
        }
    }

    cout << "\nTotal Profit: " << totalProfit << endl;
}

int main() {
    int n;
    cout << "Enter number of jobs: ";
    cin >> n;

    int jobs[n], deadline[n], profit[n];
    cout << "Enter job id, deadline and profit for each job:\n";
    for (int i = 0; i < n; i++) {
        cin >> jobs[i] >> deadline[i] >> profit[i];
    }

    jobScheduling(n, jobs, deadline, profit);

    return 0;
}
```