Assignment 6

Data Structure and Algorithms

Tapas Ranjan Nayak (20CS01064)

Processor scheduling problem:

1 processor and N jobs:

Let's start with 1 processor and N jobs then we have to process each job in that processor.

2 processor and N jobs:

Approach 1 : we first try out all possible N jobs in processor 1 and rest in processor 2 .
Approach 2 : we try to associate a value (profit) in each job if it is done on processor 1 **over** 2. Then rather than trying out all possible N jobs in processor 1 we greedly pick the most optimal (greater profit) N jobs and associate them to processor 1 and rest to 2. It turns out that function is simple: it takes the time for the job in processor 2 and processor 1 and outputs the difference between them.

3 processor and 3N jobs:

Approach 1 : we try out all possible assignments and pick the one with the optimal answer.
Approach 2 : we try out all possible N jobs on processor 1 and deduce the problem to 2 processor and 2N jobs problem.
Approach 3 : we try to develop a function that associates the profit of each job to perform them on each processor over other processors.Then we take N most profitable jobs and assign to processor 1 and deduce the problem to 2 processor and 2N jobs.

N processor and N*N jobs:

Approach 1 : we try out all possible assignments and pick the one with the optimal answer.
Approach 2 : we try out all possible n jobs on processor 1 and deduce the problem to N-1 processor and N*(N-1) jobs problem.

Example for 3 processor and 3*N jobs:

JOBS	Processor 1	Processor 2	Processor 3
1	3	3	4
2	3	4	5
3	8	7	6

(profit is time save if we perform task in process 2 over process 3)

Let's take processor 1 assign to job 1

JOBS	Processor 2	Processor 3	profit
2	4	5	1
3	7	6	-1

Total Time : 3 + 4 + 6 = 13

Let's take processor 1 assign to job 2

JOBS	Processor 2	Processor 3	profit
1	3	4	1
3	7	6	-1

Total Time : 3 + 3 + 6 = 12

Let's take processor 1 assign to job 3

JOBS	Processor 2	Processor 3	profit
1	3	4	1
2	4	5	1

Total Time : 8 + 3 + 5 = 16