

# 70068 Scheduling and Resource Allocation CW

Ali Abbas

November 27, 2024

## 1 Question 1

Everything in this section will use 0-based indexing.

### 1.1 LCL Proof

The *Least Cost Last (LCL)* rule solves the scheduling problem  $1|prec|f_{\max}$  optimally, in  $O(n^2)$  time, by constructing an optimal schedule from back to front.

#### Definitions

- $N = \{1, 2, \dots, n\}$  is the index set of all jobs
- $L \subseteq N$  is the subset of jobs without successors (which can be scheduled last)
- $p(S) = \sum_{j \in S} p_j$  is the total processing time of a subset  $S$
- $f_{\max}^*(S)$  is the cost of the optimal schedule for subset  $S$
- $f_{\max}$  denotes the maximum (not necessarily optimal) cost across all jobs in a schedule
- $f_j(p(N))$  is a cost function which assigns a penalty/cost depending on job  $j$  and the time at which the machine has finished processing,  $p(N)$ 
  - In the notation of the coursework specification  $g_j(\cdot) = f_j(\cdot)$
  - In the next section we'll use  $g_j(C_j) = T_j = \max(0, C_j - d_j)$  (tardiness)

#### Proof and Discussion

1. One job in  $L$  must be scheduled last. A job can be selected for the final job which minimises  $f_j(p(N))$ , but it can never result in a cost lower than the cost of the optimal schedule, otherwise that would be the optimal schedule. This is expressed as such:

$$f_{\max}^*(N) \geq \min_{j \in L} f_j(p(N))$$

2. And removing a job  $j \in N$  can't increase the optimal cost any more, since the cost of scheduling a job last is  $\geq 0$ , so omitting it cannot result in a more expensive schedule:

$$f_{\max}^*(N) \geq f_{\max}^*(N - \{j\}), \quad \forall j \in N$$

3. We can then select a  $J_l \in L$  that minimises  $f_j(p(N))$ :

$$f_l(p(N)) = \min_{j \in L} f_j(p(N))$$

Which gives us:

$$f_{\max}^*(N) \geq \max\{f_l(p(N)), f_{\max}^*(N - \{l\})\}$$

4. The right hand side of the above inequality is the cost of an optimal schedule, where  $J_l$  is processed last, so you can recursively apply the LCL rule to  $N - \{J_l\}$  and construct a schedule in reverse order
  - Since  $J_l$  is found in  $O(n)$  time, then with  $n$ , repeated applications of the LCL rule yields the optimal schedule in  $O(n^2)$  time

You can also show that *LCL* is optimal by using the *adjacent pairwise interchange argument*:

- You take an optimal schedule  $S$  and assume it isn't an *LCL* schedule

- Since it's not an *LCL* schedule, there must exist at least one pair where a lower cost job precedes a higher cost one
- It can then be shown that swapping these jobs yields a lower cost schedule, which is a contradiction since we took an optimal schedule
- So by virtue of the fact that a lower cost job can precede a higher cost one leads to a contradiction
- And so such pairs cannot exist for an optimal schedule, so we can reject the assumption that this isn't an *LCL* schedule
- And by contradiction then we have that *LCL* is optimal for  $1|prec|f_{\max}$

### Small Example

Example Setup:

Jobs: J0, J1, J2, J3, J4

Processing Times:  $p=[2,3,1,2,3]$

Due Dates:  $d=[6,5,7,4,9]$

Precedence Constraints (DAG):  $J0 \rightarrow J1, J0 \rightarrow J2, J3 \rightarrow J4$

Cost Function:  $f_j(C_j) = T_j = \max(0, C_j - d_j)$

**Iterations** We can schedule jobs in reverse order using the *Least Cost Last (LCL)* rule:

0. Available jobs:  $V = \{1, 2, 4\}$  (no successors)  
 $p(N) = 11$

$$f_1(p(N)) = \max(0, 11 - 5) = 6, \quad f_2(p(N)) = \max(0, 11 - 4) = 7, \quad f_4(p(N)) = \max(0, 11 - 9) = 2$$

Select  $J_4$  (minimises  $f_j(p(N))$ )

Partial schedule cost: 2

Updated schedule: [4]

1. Available jobs:  $V = \{1, 2, 3\}$  (no successors)  
 $p(N) = 8$

$$f_1(p(N)) = \max(0, 8 - 5) = 3, \quad f_2(p(N)) = \max(0, 8 - 4) = 4, \quad f_3(p(N)) = \max(0, 8 - 7) = 1$$

Select  $J_3$  (minimises  $f_j(p(N))$ )

Partial schedule cost: 2

Updated schedule: [3, 4]

2. Available jobs:  $V = \{1, 2\}$  (no successors)  
 $p(N) = 6$

$$f_1(p(N)) = \max(0, 6 - 5) = 1, \quad f_2(p(N)) = \max(0, 6 - 4) = 2$$

Select  $J_1$  (minimises  $f_j(p(N))$ )

Partial schedule cost: 2

Updated schedule: [1, 3, 4]

3. Available jobs:  $V = \{2\}$  (no successors)  
 $p(N) = 3$

$$f_2(p(N)) = \max(0, 3 - 4) = 0$$

Select  $J_2$

Partial schedule cost: 2

Updated schedule: [2, 1, 3, 4]

4. Available jobs:  $V = \{0\}$  (no successors)  
 $p(N) = 2$

$$f_0(p(N)) = \max(0, 2 - 6) = 0$$

Select  $J_0$

Partial schedule cost: 2

Updated schedule: [0, 2, 1, 3, 4]

## Results

- Final schedule:  $[J_0, J_2, J_1, J_3, J_4]$

- Completion times:

$$C_0 = 2, C_2 = 3, C_1 = 6, C_3 = 8, C_4 = 11$$

- Tardiness:

$$T_0 = 0, T_2 = 0, T_1 = 1, T_3 = 1, T_4 = 2$$

- Maximum cost:

$$f_{\max} = \max(T_0, T_2, T_1, T_3, T_4) = 2$$

## 1.2 LCL Implementation

### Iterations

0. Available jobs:  $V = \{0, 30\}$  (no successors)

$$p(N) = 170$$

$$f_0(p(N)) = \max(0, 170 - 172) = 0, \quad f_{30}(p(N)) = \max(0, 170 - 269) = 0$$

Select  $J_{30}$  (minimises  $f_j(p(N))$ )

Partial schedule cost: 0

Updated schedule:  $[30]$

1. Available jobs:  $V = \{0, 1, 10\}$  (no successors)

$$p(N) = 160$$

$$f_0(p(N)) = \max(0, 160 - 172) = 0, \quad f_1(p(N)) = \max(0, 160 - 82) = 78, \quad f_{10}(p(N)) = \max(0, 160 - 253) = 0$$

Select  $J_0$  (minimises  $f_j(p(N))$ )

Partial schedule cost: 0

Updated schedule:  $[0, 30]$

...

4. Available jobs:  $V = \{1, 4\}$  (no successors)

$$p(N) = 147$$

$$f_1(p(N)) = \max(0, 147 - 82) = 65, \quad f_4(p(N)) = \max(0, 147 - 93) = 54$$

Select  $J_1$  (minimises  $f_j(p(N))$ ) Partial schedule cost: 65

Updated schedule:  $[1, 14, 10, 0, 30]$

This iteration was notable since it incurred the maximum tardiness incurred for this problem.

...

17. Available jobs:  $V = \{17, 18, 20, 5\}$  (no successors)

$$p(N) = 68$$

$$f_{17}(p(N)) = \max(0, 68 - 77) = 0, \quad f_{18}(p(N)) = \max(0, 68 - 88) = 0,$$

$$f_{20}(p(N)) = \max(0, 68 - 71) = 0, \quad f_5(p(N)) = \max(0, 68 - 71) = 0$$

Select  $J_{17}$  (arbitrarily breaking the tie)

Partial schedule cost: 65

Updated schedule:  $[17, 16, 15, 13, 28, 12, 27, 26, 25, 24, 23, 11, 4, 1, 14, 10, 0, 30]$

...

30. Available jobs:  $V = \{29\}$  (no successors)

$$p(N) = 2$$

$$f_{29}(p(N)) = \max(0, 2 - 329) = 0$$

Select  $J_{29}$  (minimises  $f_j(p(N))$ )

Final schedule cost: 65

Final schedule:  $[29, 9, 8, 3, 2, 7, 6, 22, 21, 5, 20, 19, 18, 17, 16, 15, 13, 28, 12, 27, 26, 25, 24, 23, 11, 4, 1, 14, 10, 0, 30]$

Note: the tie breaking doesn't specify any ordering in the current implementation (since the set of available jobs is backed by a hash table).

## 2 Question 2

### 2.1 Tabu Search

Throughout this section, I used  $\gamma = 10$  and  $L = 20$ . I forced the initial solution to be the one specified in the coursework specification.

I also included the first few iterations and all notable intermediate solutions for each value of  $K$ .

#### 2.1.1 K=10

##### Iterations

0. **Candidate schedule costs:** [341, 341, 341, 341, 341, 341, 338, 343, 341, 341, 341, 339, 351, 331, 359, 341, 336, 346, 336, 346, 336]  
**Tabu list:** { (17, 25) }  
**Best cost so far:** 331  
**Tabu list used?:** No  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 24, 16, 14, 5, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 331
1. **Candidate schedule costs:** [331, 331, 331, 331, 331, 331, 328, 333, 331, 331, 331, 329, 341, 331, 326, 336, 326, 336, 326]  
**Tabu list:** { (17, 25), (5, 14) }  
**Best cost so far:** 326  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 24, 16, 5, 14, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 326
2. **Candidate schedule costs:** [326, 326, 326, 326, 326, 323, 328, 326, 326, 326, 324, 336, 326, 320, 331, 326, 321, 331, 321, 326]  
**Tabu list:** { (17, 25), (5, 14), (5, 16) }  
**Best cost so far:** 320  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 24, 5, 16, 14, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 320
3. **Candidate schedule costs:** [320, 320, 320, 320, 317, 322, 320, 320, 320, 318, 330, 302, 326, 320, 315, 325, 315, 320, 320]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24) }  
**Best cost so far:** 302  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 5, 24, 16, 14, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 302
4. **Candidate schedule costs:** [302, 302, 302, 302, 299, 304, 302, 302, 302, 300, 312, 292, 320, 302, 302, 297, 307, 297, 302, 302]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25) }  
**Best cost so far:** 292  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 14, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 292
5. **Candidate schedule costs:** [292, 292, 292, 289, 294, 292, 292, 292, 290, 288, 302, 292, 292, 287, 297, 287, 292, 292, 292]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23) }  
**Best cost so far:** 287  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 14, 15, 23, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 287

6. **Candidate schedule costs:** [287, 287, 287, 284, 289, 287, 287, 287, 285, 283, 297, 287, 282, 292, 292, 282, 287, 287, 287]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15) }  
**Best cost so far:** 282  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 15, 14, 23, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 282
7. **Candidate schedule costs:** [282, 282, 282, 279, 284, 282, 282, 282, 280, 278, 292, 282, 287, 282, 287, 277, 282, 282, 282]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10) }  
**Best cost so far:** 277  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 277
8. **Candidate schedule costs:** [277, 277, 274, 279, 277, 277, 277, 275, 273, 287, 277, 282, 277, 282, 277, 277, 277, 277]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17) }  
**Best cost so far:** 273  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 273
9. **Candidate schedule costs:** [273, 270, 275, 273, 273, 273, 271, 273, 277, 283, 273, 278, 273, 278, 273, 273, 273, 273]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21) }  
**Best cost so far:** 270  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 2, 21, 26, 27, 7, 6, 18, 20, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 270

**Results** There weren't enough iterations to reach an optimal solution, evidenced by the fact that every iteration yielded a more optimal solution.

### 2.1.2 K=100

#### Iterations

0. **Candidate schedule costs:** [341, 341, 341, 341, 341, 341, 338, 343, 341, 341, 341, 339, 351, 331, 359, 341, 336, 346, 336, 346, 336]  
**Tabu list:** { (17, 25) }  
**Best cost so far:** 331  
**Tabu list used?:** No  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 24, 16, 14, 5, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 331
1. **Candidate schedule costs:** [331, 331, 331, 331, 331, 331, 328, 333, 331, 331, 331, 329, 341, 331, 326, 336, 326, 336, 326]  
**Tabu list:** { (17, 25), (5, 14) }  
**Best cost so far:** 326  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 24, 16, 5, 14, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 326
2. **Candidate schedule costs:** [326, 326, 326, 326, 326, 323, 328, 326, 326, 326, 324, 336, 326, 320, 331, 326, 321, 331, 321, 326]  
**Tabu list:** { (17, 25), (5, 14), (5, 16) }  
**Best cost so far:** 320  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 24, 5, 16, 14, 23, 15, 4, 10, 1, 0, 30]

- 10, 1, 0, 30]  
**Selected neighbour cost:** 320
3. **Candidate schedule costs:** [320, 320, 320, 320, 317, 322, 320, 320, 320, 318, 330, 302, 326, 320, 315, 325, 315, 320, 320]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24) }  
**Best cost so far:** 302  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 5, 24, 16, 14, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 302
4. **Candidate schedule costs:** [302, 302, 302, 302, 299, 304, 302, 302, 302, 300, 312, 292, 320, 302, 302, 297, 307, 297, 302, 302]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25) }  
**Best cost so far:** 292  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 14, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 292
5. **Candidate schedule costs:** [292, 292, 292, 289, 294, 292, 292, 292, 290, 288, 302, 292, 292, 287, 297, 287, 292, 292, 292]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23) }  
**Best cost so far:** 287  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 14, 15, 23, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 287
6. **Candidate schedule costs:** [287, 287, 287, 284, 289, 287, 287, 287, 285, 283, 297, 287, 282, 292, 292, 282, 287, 287, 287]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15) }  
**Best cost so far:** 282  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 15, 14, 23, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 282
7. **Candidate schedule costs:** [282, 282, 282, 279, 284, 282, 282, 282, 280, 278, 292, 282, 287, 282, 287, 277, 282, 282, 282]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10) }  
**Best cost so far:** 277  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 277
8. **Candidate schedule costs:** [277, 277, 274, 279, 277, 277, 277, 275, 273, 287, 277, 282, 277, 282, 277, 277, 277, 277]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17) }  
**Best cost so far:** 273  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 273
9. **Candidate schedule costs:** [273, 270, 275, 273, 273, 273, 271, 273, 277, 283, 273, 278, 273, 278, 273, 273, 273, 273, 273]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21) }  
**Best cost so far:** 270  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 2, 21, 26, 27, 7, 6, 18, 20, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 270

10. **Candidate schedule costs:** [273, 270, 270, 270, 270, 268, 270, 274, 280, 270, 275, 270, 275, 270, 270, 270, 270, 267]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19) }  
**Best cost so far:** 267  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 7, 6, 18, 20, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 267
11. **Candidate schedule costs:** [267, 267, 267, 267, 265, 267, 271, 277, 267, 272, 267, 272, 267, 267, 267, 267, 270, 267]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19), (18, 20) }  
**Best cost so far:** 265  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 7, 6, 20, 18, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 265
12. **Candidate schedule costs:** [265, 265, 259, 267, 263, 269, 275, 265, 270, 265, 270, 265, 265, 265, 265, 268, 265, 265]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19), (18, 20), (6, 20) }  
**Best cost so far:** 259  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 7, 20, 6, 18, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 259
13. **Candidate schedule costs:** [259, 259, 265, 259, 257, 263, 269, 259, 264, 259, 264, 259, 259, 259, 259, 262, 259, 259, 259]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19), (18, 20), (6, 20), (5, 18) }  
**Best cost so far:** 257  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 7, 20, 6, 5, 18, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 257
16. **Candidate schedule costs:** [259, 267, 257, 257, 239, 262, 257, 262, 257, 257, 257, 257, 260, 257, 257, 257, 257, 257]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19), (18, 20), (6, 20), (5, 18), (7, 20), (16, 24), (15, 24) }  
**Best cost so far:** 239  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 20, 7, 6, 5, 18, 17, 25, 16, 15, 24, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 239
18. **Candidate schedule costs:** [239, 229, 257, 239, 239, 244, 239, 239, 239, 239, 242, 239, 239, 239, 239, 239, 241]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19), (18, 20), (6, 20), (5, 18), (7, 20), (16, 24), (15, 24), (16, 25), (15, 25) }  
**Best cost so far:** 229  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 20, 7, 6, 5, 18, 17, 16, 15, 25, 24, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 229
36. **Candidate schedule costs:** [229, 229, 223, 232, 229, 229, 229, 229, 229, 229, 231, 234, 229, 234, 229, 229, 229, 229, 229]  
**Tabu list:** { (16, 25), (15, 25), (14, 24), (14, 25), (22, 28), (9, 28), (9, 22), (8, 28), (8, 22), (13, 28), (13, 22), (8, 13), (9, 13), (12, 28), (12, 22), (8, 12), (11, 28), (11, 22), (3, 28), (2, 28) }  
**Best cost so far:** 223  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 13, 9, 12, 8, 11, 22, 3, 2, 28, 19, 21, 26, 27, 20, 7, 6, 5, 18, 17, 16, 15, 14, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 223

45. **Candidate schedule costs:** [223, 221, 225, 228, 223, 228, 223, 223, 223, 223, 223, 223, 226, 223, 223, 223, 223]  
**Tabu list:** { (13, 28), (13, 22), (8, 13), (9, 13), (12, 28), (12, 22), (8, 12), (11, 28), (11, 22), (3, 28), (2, 28), (19, 28), (21, 28), (19, 21), (26, 27), (20, 26), (20, 27), (7, 26), (6, 26), (5, 26) }  
**Best cost so far:** 221  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 13, 9, 12, 8, 11, 22, 3, 2, 21, 19, 28, 20, 27, 7, 6, 5, 26, 18, 17, 16, 15, 14, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 221
47. **Candidate schedule costs:** [221, 219, 226, 221, 226, 221, 221, 221, 221, 221, 221, 224, 221, 221, 221, 221, 221]  
**Tabu list:** { (8, 13), (9, 13), (12, 28), (12, 22), (8, 12), (11, 28), (11, 22), (3, 28), (2, 28), (19, 28), (21, 28), (19, 21), (26, 27), (20, 26), (20, 27), (7, 26), (6, 26), (5, 26), (18, 26), (17, 26) }  
**Best cost so far:** 219  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 13, 9, 12, 8, 11, 22, 3, 2, 21, 19, 28, 20, 27, 7, 6, 5, 18, 17, 26, 16, 15, 14, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 219
49. **Candidate schedule costs:** [219, 217, 224, 219, 224, 219, 219, 219, 219, 219, 219, 222, 219, 219, 219, 219, 219]  
**Tabu list:** { (12, 28), (12, 22), (8, 12), (11, 28), (11, 22), (3, 28), (2, 28), (19, 28), (21, 28), (19, 21), (26, 27), (20, 26), (20, 27), (7, 26), (6, 26), (5, 26), (18, 26), (17, 26), (16, 26), (15, 26) }  
**Best cost so far:** 217  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 13, 9, 12, 8, 11, 22, 3, 2, 21, 19, 28, 20, 27, 7, 6, 5, 18, 17, 16, 15, 26, 14, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 217
55. **Candidate schedule costs:** [222, 217, 217, 217, 217, 211, 220, 217, 217, 217, 217, 217, 219, 222, 217]  
**Tabu list:** { (2, 28), (19, 28), (21, 28), (19, 21), (26, 27), (20, 26), (20, 27), (7, 26), (6, 26), (5, 26), (18, 26), (17, 26), (16, 26), (15, 26), (14, 26), (9, 13), (8, 11), (8, 22), (3, 8), (2, 8) }  
**Best cost so far:** 211  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 9, 13, 12, 11, 22, 3, 2, 8, 21, 19, 28, 20, 27, 7, 6, 5, 18, 17, 16, 15, 14, 26, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 211
61. **Candidate schedule costs:** [211, 211, 211, 211, 209, 217, 211, 211, 211, 211, 211, 211, 213, 216, 211, 216]  
**Tabu list:** { (20, 27), (7, 26), (6, 26), (5, 26), (18, 26), (17, 26), (16, 26), (15, 26), (14, 26), (9, 13), (8, 11), (8, 22), (3, 8), (2, 8), (11, 22), (12, 22), (13, 22), (9, 22), (3, 11), (2, 11) }  
**Best cost so far:** 209  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 22, 9, 13, 12, 3, 2, 11, 8, 21, 19, 28, 20, 27, 7, 6, 5, 18, 17, 16, 15, 14, 26, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 209
63. **Candidate schedule costs:** [209, 209, 206, 211, 209, 209, 209, 209, 209, 209, 209, 211, 214, 209, 214, 209, 209]  
**Tabu list:** { (6, 26), (5, 26), (18, 26), (17, 26), (16, 26), (15, 26), (14, 26), (9, 13), (8, 11), (8, 22), (3, 8), (2, 8), (11, 22), (12, 22), (13, 22), (9, 22), (3, 11), (2, 11), (3, 12), (2, 12) }  
**Best cost so far:** 206  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 22, 9, 13, 3, 2, 12, 11, 8, 21, 19, 28, 20, 27, 7, 6, 5, 18, 17, 16, 15, 14, 26, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 206
76. **Candidate schedule costs:** [208, 208, 206, 201, 211, 206, 206, 206, 209, 206, 206, 206, 206, 206, 206, 206]  
**Tabu list:** { (12, 22), (13, 22), (9, 22), (3, 11), (2, 11), (3, 12), (2, 12), (8, 21), (11, 21), (12, 21), (8, 19), (11, 19), (8, 28), (8, 20), (8, 27), (14, 26), (14, 25), (14, 24), (14, 23), (4, 14) }  
**Best cost so far:** 201  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 22, 9, 13, 3, 2, 21, 12, 19, 11, 28, 20, 27, 8, 7, 6, 5, 18, 17, 16, 15, 26, 25, 24, 23, 4, 14, 1, 10, 0, 30]  
**Selected neighbour cost:** 201



77. **Candidate schedule costs:** [203, 206, 196, 206, 201, 201, 201, 204, 201, 201, 201, 201, 201, 201, 203]  
**Tabu list:** { (13, 22), (9, 22), (3, 11), (2, 11), (3, 12), (2, 12), (8, 21), (11, 21), (12, 21), (8, 19), (11, 19), (8, 28), (8, 20), (8, 27), (14, 26), (14, 25), (14, 24), (14, 23), (4, 14), (1, 14) }  
**Best cost so far:** 196  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 22, 9, 13, 3, 2, 21, 12, 19, 11, 28, 20, 27, 8, 7, 6, 5, 18, 17, 16, 15, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]  
**Selected neighbour cost:** 196
82. **Candidate schedule costs:** [201, 196, 196, 196, 186, 199, 196, 196, 196, 196, 196, 196, 198, 198]  
**Tabu list:** { (2, 12), (8, 21), (11, 21), (12, 21), (8, 19), (11, 19), (8, 28), (8, 20), (8, 27), (14, 26), (14, 25), (14, 24), (14, 23), (4, 14), (1, 14), (9, 13), (13, 22), (9, 22), (3, 22), (2, 22) }  
**Best cost so far:** 186  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 13, 9, 3, 2, 22, 21, 12, 19, 11, 28, 20, 27, 8, 7, 6, 5, 18, 17, 16, 15, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]  
**Selected neighbour cost:** 186
99. **Candidate schedule costs:** [186, 186, 188, 188, 191, 186, 186, 186, 186, 186, 186]  
**Tabu list:** { (9, 22), (3, 22), (2, 22), (3, 9), (3, 13), (2, 9), (2, 13), (12, 21), (12, 22), (9, 12), (19, 21), (19, 22), (9, 19), (12, 19), (11, 21), (11, 22), (9, 11), (21, 28), (22, 28), (20, 27) }  
**Best cost so far:** 186  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 3, 2, 13, 19, 12, 11, 9, 28, 22, 21, 27, 20, 8, 7, 6, 5, 18, 17, 16, 15, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]  
**Selected neighbour cost:** 186

**Results** There were again not enough iterations to find the optimal solution. The best solution found was 186, which remained the same for more than 10 iterations, indicating that there was evidence of cycling for this value of  $K$ .

### 2.1.3 K=1000

#### Iterations

0. **Candidate schedule costs:** [341, 341, 341, 341, 341, 341, 338, 343, 341, 341, 341, 339, 351, 331, 359, 341, 336, 346, 336, 346, 336]  
**Tabu list:** { (17, 25) }  
**Best cost so far:** 331  
**Tabu list used?:** No  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 24, 16, 14, 5, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 331
1. **Candidate schedule costs:** [331, 331, 331, 331, 331, 331, 328, 333, 331, 331, 331, 329, 341, 331, 326, 336, 326, 336, 326]  
**Tabu list:** { (17, 25), (5, 14) }  
**Best cost so far:** 326  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 24, 16, 5, 14, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 326
2. **Candidate schedule costs:** [326, 326, 326, 326, 326, 323, 328, 326, 326, 326, 324, 336, 326, 320, 331, 326, 321, 331, 321, 326]  
**Tabu list:** { (17, 25), (5, 14), (5, 16) }  
**Best cost so far:** 320  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 24, 5, 16, 14, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 320
3. **Candidate schedule costs:** [320, 320, 320, 320, 317, 322, 320, 320, 320, 318, 330, 302, 326, 320, 315, 325, 315, 320, 320]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24) }  
**Best cost so far:** 302  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 25, 5, 24, 16, 14, 23, 15, 4,

- 10, 1, 0, 30]  
**Selected neighbour cost:** 302
4. **Candidate schedule costs:** [302, 302, 302, 302, 299, 304, 302, 302, 302, 300, 312, 292, 320, 302, 302, 297, 307, 297, 302, 302]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25) }  
**Best cost so far:** 292  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 14, 23, 15, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 292
5. **Candidate schedule costs:** [292, 292, 292, 289, 294, 292, 292, 292, 290, 288, 302, 292, 292, 287, 297, 287, 292, 292, 292]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23) }  
**Best cost so far:** 287  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 14, 15, 23, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 287
6. **Candidate schedule costs:** [287, 287, 287, 284, 289, 287, 287, 287, 285, 283, 297, 287, 282, 292, 292, 282, 287, 287, 287]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15) }  
**Best cost so far:** 282  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 15, 14, 23, 4, 10, 1, 0, 30]  
**Selected neighbour cost:** 282
7. **Candidate schedule costs:** [282, 282, 282, 279, 284, 282, 282, 282, 280, 278, 292, 282, 287, 282, 287, 277, 282, 282, 282]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10) }  
**Best cost so far:** 277  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 17, 5, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 277
8. **Candidate schedule costs:** [277, 277, 274, 279, 277, 277, 277, 275, 273, 287, 277, 282, 277, 282, 277, 277, 277, 277]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17) }  
**Best cost so far:** 273  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 21, 2, 26, 27, 7, 6, 18, 20, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 273
9. **Candidate schedule costs:** [273, 270, 275, 273, 273, 273, 271, 273, 277, 283, 273, 278, 273, 278, 273, 273, 273, 273, 273]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21) }  
**Best cost so far:** 270  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 19, 2, 21, 26, 27, 7, 6, 18, 20, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 270
10. **Candidate schedule costs:** [273, 270, 270, 270, 270, 268, 270, 274, 280, 270, 275, 270, 275, 270, 270, 270, 270, 270, 267]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19) }  
**Best cost so far:** 267  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 7, 6, 18, 20, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 267

11. **Candidate schedule costs:** [267, 267, 267, 267, 265, 267, 271, 277, 267, 272, 267, 272, 267, 267, 267, 267, 270, 267]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19), (18, 20) }  
**Best cost so far:** 265  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 7, 6, 20, 18, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 265
12. **Candidate schedule costs:** [265, 265, 259, 267, 263, 269, 275, 265, 270, 265, 270, 265, 265, 265, 265, 268, 265, 265]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19), (18, 20), (6, 20) }  
**Best cost so far:** 259  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 7, 20, 6, 18, 5, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 259
13. **Candidate schedule costs:** [259, 259, 265, 259, 257, 263, 269, 259, 264, 259, 264, 259, 259, 259, 259, 262, 259, 259, 259]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19), (18, 20), (6, 20), (5, 18) }  
**Best cost so far:** 257  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 7, 20, 6, 5, 18, 17, 25, 24, 16, 15, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 257
16. **Candidate schedule costs:** [259, 267, 257, 257, 239, 262, 257, 262, 257, 257, 257, 257, 260, 257, 257, 257, 257, 257]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19), (18, 20), (6, 20), (5, 18), (7, 20), (16, 24), (15, 24) }  
**Best cost so far:** 239  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 20, 7, 6, 5, 18, 17, 25, 16, 15, 24, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 239
18. **Candidate schedule costs:** [239, 229, 257, 239, 239, 244, 239, 239, 239, 239, 242, 239, 239, 239, 239, 239, 241]  
**Tabu list:** { (17, 25), (5, 14), (5, 16), (5, 24), (5, 25), (15, 23), (14, 15), (1, 10), (5, 17), (2, 21), (2, 19), (18, 20), (6, 20), (5, 18), (7, 20), (16, 24), (15, 24), (16, 25), (15, 25) }  
**Best cost so far:** 229  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 28, 22, 9, 8, 13, 12, 11, 3, 2, 19, 21, 26, 27, 20, 7, 6, 5, 18, 17, 16, 15, 25, 24, 14, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 229
36. **Candidate schedule costs:** [229, 229, 223, 232, 229, 229, 229, 229, 229, 229, 231, 234, 229, 234, 229, 229, 229, 229, 229]  
**Tabu list:** { (16, 25), (15, 25), (14, 24), (14, 25), (22, 28), (9, 28), (9, 22), (8, 28), (8, 22), (13, 28), (13, 22), (8, 13), (9, 13), (12, 28), (12, 22), (8, 12), (11, 28), (11, 22), (3, 28), (2, 28) }  
**Best cost so far:** 223  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 13, 9, 12, 8, 11, 22, 3, 2, 28, 19, 21, 26, 27, 20, 7, 6, 5, 18, 17, 16, 15, 14, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 223
45. **Candidate schedule costs:** [223, 221, 225, 228, 223, 228, 223, 223, 223, 223, 223, 223, 226, 223, 223, 223, 223, 223]  
**Tabu list:** { (13, 28), (13, 22), (8, 13), (9, 13), (12, 28), (12, 22), (8, 12), (11, 28), (11, 22), (3, 28), (2, 28), (19, 28), (21, 28), (19, 21), (26, 27), (20, 26), (20, 27), (7, 26), (6, 26), (5, 26) }  
**Best cost so far:** 221  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 13, 9, 12, 8, 11, 22, 3, 2, 21, 19, 28, 20, 27, 7, 6, 5, 26, 18, 17, 16, 15, 14, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 221

47. **Candidate schedule costs:** [221, 219, 226, 221, 226, 221, 221, 221, 221, 221, 221, 224, 221, 221, 221, 221, 221]  
**Tabu list:** { (8, 13), (9, 13), (12, 28), (12, 22), (8, 12), (11, 28), (11, 22), (3, 28), (2, 28), (19, 28), (21, 28), (19, 21), (26, 27), (20, 26), (20, 27), (7, 26), (6, 26), (5, 26), (18, 26), (17, 26) }  
**Best cost so far:** 219  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 13, 9, 12, 8, 11, 22, 3, 2, 21, 19, 28, 20, 27, 7, 6, 5, 18, 17, 26, 16, 15, 14, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 219
49. **Candidate schedule costs:** [219, 217, 224, 219, 224, 219, 219, 219, 219, 219, 219, 222, 219, 219, 219, 219, 219, 221]  
**Tabu list:** { (12, 28), (12, 22), (8, 12), (11, 28), (11, 22), (3, 28), (2, 28), (19, 28), (21, 28), (19, 21), (26, 27), (20, 26), (20, 27), (7, 26), (6, 26), (5, 26), (18, 26), (17, 26), (16, 26), (15, 26) }  
**Best cost so far:** 217  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 13, 9, 12, 8, 11, 22, 3, 2, 21, 19, 28, 20, 27, 7, 6, 5, 18, 17, 16, 15, 26, 14, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 217
55. **Candidate schedule costs:** [222, 217, 217, 217, 217, 211, 220, 217, 217, 217, 217, 217, 219, 222, 217]  
**Tabu list:** { (2, 28), (19, 28), (21, 28), (19, 21), (26, 27), (20, 26), (20, 27), (7, 26), (6, 26), (5, 26), (18, 26), (17, 26), (16, 26), (15, 26), (14, 26), (9, 13), (8, 11), (8, 22), (3, 8), (2, 8) }  
**Best cost so far:** 211  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 9, 13, 12, 11, 22, 3, 2, 8, 21, 19, 28, 20, 27, 7, 6, 5, 18, 17, 16, 15, 14, 26, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 211
61. **Candidate schedule costs:** [211, 211, 211, 211, 209, 217, 211, 211, 211, 211, 211, 211, 211, 213, 216, 211, 216]  
**Tabu list:** { (20, 27), (7, 26), (6, 26), (5, 26), (18, 26), (17, 26), (16, 26), (15, 26), (14, 26), (9, 13), (8, 11), (8, 22), (3, 8), (2, 8), (11, 22), (12, 22), (13, 22), (9, 22), (3, 11), (2, 11) }  
**Best cost so far:** 209  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 22, 9, 13, 12, 3, 2, 11, 8, 21, 19, 28, 20, 27, 7, 6, 5, 18, 17, 16, 15, 14, 26, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 209
63. **Candidate schedule costs:** [209, 209, 206, 211, 209, 209, 209, 209, 209, 209, 209, 211, 214, 209, 214, 209, 209]  
**Tabu list:** { (6, 26), (5, 26), (18, 26), (17, 26), (16, 26), (15, 26), (14, 26), (9, 13), (8, 11), (8, 22), (3, 8), (2, 8), (11, 22), (12, 22), (13, 22), (9, 22), (3, 11), (2, 11), (3, 12), (2, 12) }  
**Best cost so far:** 206  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 22, 9, 13, 3, 2, 12, 11, 8, 21, 19, 28, 20, 27, 7, 6, 5, 18, 17, 16, 15, 14, 26, 25, 24, 23, 4, 1, 10, 0, 30]  
**Selected neighbour cost:** 206
76. **Candidate schedule costs:** [208, 208, 206, 201, 211, 206, 206, 206, 209, 206, 206, 206, 206, 206, 206, 206]  
**Tabu list:** { (12, 22), (13, 22), (9, 22), (3, 11), (2, 11), (3, 12), (2, 12), (8, 21), (11, 21), (12, 21), (8, 19), (11, 19), (8, 28), (8, 20), (8, 27), (14, 26), (14, 25), (14, 24), (14, 23), (4, 14) }  
**Best cost so far:** 201  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 22, 9, 13, 3, 2, 21, 12, 19, 11, 28, 20, 27, 8, 7, 6, 5, 18, 17, 16, 15, 26, 25, 24, 23, 4, 14, 1, 10, 0, 30]  
**Selected neighbour cost:** 201
77. **Candidate schedule costs:** [203, 206, 196, 206, 201, 201, 201, 204, 201, 201, 201, 201, 201, 201, 201, 201, 203]  
**Tabu list:** { (13, 22), (9, 22), (3, 11), (2, 11), (3, 12), (2, 12), (8, 21), (11, 21), (12, 21), (8, 19), (11, 19), (8, 28), (8, 20), (8, 27), (14, 26), (14, 25), (14, 24), (14, 23), (4, 14), (1, 14) }  
**Best cost so far:** 196  
**Tabu list used?:** Yes  
**Selected neighbour:** [29, 22, 9, 13, 3, 2, 21, 12, 19, 11, 28, 20, 27, 8, 7, 6, 5, 18, 17, 16, 15, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]  
**Selected neighbour cost:** 196
82. **Candidate schedule costs:** [201, 196, 196, 196, 186, 199, 196, 196, 196, 196, 196, 196, 196, 198, 198]  
**Tabu list:** { (2, 12), (8, 21), (11, 21), (12, 21), (8, 19), (11, 19), (8, 28), (8, 20), (8, 27), (14, 26), (14, 25), (14, 24),

- (14, 23), (4, 14), (1, 14), (9, 13), (13, 22), (9, 22), (3, 22), (2, 22) }
- Best cost so far:** 186
- Tabu list used?:** Yes
- Selected neighbour:** [29, 13, 9, 3, 2, 22, 21, 12, 19, 11, 28, 20, 27, 8, 7, 6, 5, 18, 17, 16, 15, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]
- Selected neighbour cost:** 186
166. **Candidate schedule costs:** [188, 184, 188, 191, 188, 186, 186, 186, 186, 186, 186]
- Tabu list:** { (11, 22), (12, 22), (13, 22), (18, 21), (19, 21), (11, 21), (12, 21), (13, 21), (11, 19), (12, 19), (11, 18), (11, 28), (11, 27), (11, 20), (9, 11), (8, 11), (7, 11), (6, 11), (5, 11), (11, 17) }
- Best cost so far:** 184
- Tabu list used?:** Yes
- Selected neighbour:** [29, 3, 2, 22, 21, 13, 19, 12, 18, 28, 27, 20, 9, 8, 7, 6, 5, 17, 11, 16, 15, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]
- Selected neighbour cost:** 184
168. **Candidate schedule costs:** [184, 182, 186, 189, 186, 184, 184, 184, 184, 184, 184, 188]
- Tabu list:** { (13, 22), (18, 21), (19, 21), (11, 21), (12, 21), (13, 21), (11, 19), (12, 19), (11, 18), (11, 28), (11, 27), (11, 20), (9, 11), (8, 11), (7, 11), (6, 11), (5, 11), (11, 17), (11, 16), (11, 15) }
- Best cost so far:** 182
- Tabu list used?:** Yes
- Selected neighbour:** [29, 3, 2, 22, 21, 13, 19, 12, 18, 28, 27, 20, 9, 8, 7, 6, 5, 17, 16, 15, 11, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]
- Selected neighbour cost:** 182
224. **Candidate schedule costs:** [182, 180, 186, 184, 182, 187, 182, 182, 185, 182, 182, 182, 182, 182, 182, 184]
- Tabu list:** { (2, 28), (3, 28), (12, 28), (13, 28), (18, 22), (19, 22), (18, 27), (19, 27), (22, 27), (9, 18), (9, 19), (9, 22), (18, 21), (19, 21), (8, 18), (8, 19), (7, 18), (18, 20), (6, 18), (5, 18) }
- Best cost so far:** 180
- Tabu list used?:** Yes
- Selected neighbour:** [29, 28, 13, 12, 3, 2, 27, 9, 22, 21, 8, 19, 7, 20, 6, 5, 18, 17, 16, 15, 11, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]
- Selected neighbour cost:** 180
253. **Candidate schedule costs:** [180, 177, 182, 182, 180, 185, 180, 180, 180, 180, 180, 180, 180, 180, 180, 180, 180]
- Tabu list:** { (2, 28), (2, 12), (2, 13), (9, 27), (9, 28), (9, 12), (9, 13), (2, 9), (22, 27), (22, 28), (12, 22), (21, 27), (21, 28), (8, 27), (8, 28), (19, 27), (7, 27), (20, 27), (6, 27), (5, 27) }
- Best cost so far:** 177
- Tabu list used?:** Yes
- Selected neighbour:** [29, 3, 9, 2, 13, 22, 12, 21, 8, 28, 19, 7, 20, 6, 5, 27, 18, 17, 16, 15, 26, 25, 24, 23, 11, 4, 1, 14, 10, 0, 30]
- Selected neighbour cost:** 177
255. **Candidate schedule costs:** [177, 174, 179, 177, 182, 177, 177, 177, 177, 177, 177, 177, 177, 177, 177, 177, 177]
- Tabu list:** { (2, 13), (9, 27), (9, 28), (9, 12), (9, 13), (2, 9), (22, 27), (22, 28), (12, 22), (21, 27), (21, 28), (8, 27), (8, 28), (19, 27), (7, 27), (20, 27), (6, 27), (5, 27), (18, 27), (17, 27) }
- Best cost so far:** 174
- Tabu list used?:** Yes
- Selected neighbour:** [29, 3, 9, 2, 13, 22, 12, 21, 8, 28, 19, 7, 20, 6, 5, 18, 17, 27, 16, 15, 26, 25, 24, 23, 11, 4, 1, 14, 10, 0, 30]
- Selected neighbour cost:** 174
257. **Candidate schedule costs:** [174, 171, 176, 174, 179, 174, 174, 174, 174, 174, 174, 174, 174, 174, 174, 174, 174]
- Tabu list:** { (9, 28), (9, 12), (9, 13), (2, 9), (22, 27), (22, 28), (12, 22), (21, 27), (21, 28), (8, 27), (8, 28), (19, 27), (7, 27), (20, 27), (6, 27), (5, 27), (18, 27), (17, 27), (16, 27), (15, 27) }
- Best cost so far:** 171
- Tabu list used?:** Yes
- Selected neighbour:** [29, 3, 9, 2, 13, 22, 12, 21, 8, 28, 19, 7, 20, 6, 5, 18, 17, 16, 15, 27, 26, 25, 24, 23, 11, 4, 1, 14, 10, 0, 30]
- Selected neighbour cost:** 171
312. **Candidate schedule costs:** [171, 170, 173, 173, 171, 171, 176, 171, 171, 171, 173, 171, 171, 171, 171, 171]
- Tabu list:** { (12, 28), (13, 28), (2, 28), (3, 28), (9, 12), (9, 13), (12, 19), (13, 19), (9, 19), (2, 19), (8, 12), (8, 13), (7, 12), (7, 13), (12, 22), (13, 22), (6, 12), (12, 21), (5, 12), (12, 20) }

- Best cost so far: 170  
Tabu list used?: Yes  
Selected neighbour: [29, 28, 3, 19, 2, 9, 8, 7, 22, 13, 6, 21, 5, 20, 12, 18, 17, 16, 15, 11, 27, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]  
Selected neighbour cost: 170
314. Candidate schedule costs: [170, 167, 172, 170, 170, 175, 170, 170, 170, 172, 170, 170, 170, 170, 170, 170, 170]  
Tabu list: { (2, 28), (3, 28), (9, 12), (9, 13), (12, 19), (13, 19), (9, 19), (2, 19), (8, 12), (8, 13), (7, 12), (7, 13), (12, 22), (13, 22), (6, 12), (12, 21), (5, 12), (12, 20), (12, 18), (12, 17) }  
Best cost so far: 167  
Tabu list used?: Yes  
Selected neighbour: [29, 28, 3, 19, 2, 9, 8, 7, 22, 13, 6, 21, 5, 20, 18, 17, 12, 16, 15, 11, 27, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]  
Selected neighbour cost: 167
316. Candidate schedule costs: [167, 164, 169, 167, 167, 172, 167, 167, 167, 169, 167, 167, 167, 167, 167, 167, 167]  
Tabu list: { (9, 12), (9, 13), (12, 19), (13, 19), (9, 19), (2, 19), (8, 12), (8, 13), (7, 12), (7, 13), (12, 22), (13, 22), (6, 12), (12, 21), (5, 12), (12, 20), (12, 18), (12, 17), (12, 16), (12, 15) }  
Best cost so far: 164  
Tabu list used?: Yes  
Selected neighbour: [29, 28, 3, 19, 2, 9, 8, 7, 22, 13, 6, 21, 5, 20, 18, 17, 16, 15, 12, 11, 27, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]  
Selected neighbour cost: 164
345. Candidate schedule costs: [164, 158, 167, 164, 164, 164, 169, 164, 164, 174, 164, 164, 164, 164, 164, 164, 164]  
Tabu list: { (2, 28), (9, 28), (2, 9), (3, 9), (9, 19), (8, 28), (7, 28), (22, 28), (7, 22), (8, 22), (13, 28), (7, 13), (6, 28), (21, 28), (5, 28), (20, 28), (18, 28), (17, 28), (16, 28), (15, 28) }  
Best cost so far: 158  
Tabu list used?: Yes  
Selected neighbour: [29, 9, 19, 3, 2, 22, 8, 13, 7, 6, 21, 5, 20, 18, 17, 16, 15, 28, 27, 12, 26, 25, 24, 23, 11, 4, 1, 14, 10, 0, 30]  
Selected neighbour cost: 158
999. Candidate schedule costs: [158, 158, 158, 161, 158, 158, 158, 163, 158, 158, 158, 158, 158]  
Tabu list: { (11, 26), (3, 18), (9, 19), (3, 9), (3, 19), (9, 18), (2, 9), (2, 18), (2, 19), (13, 21), (5, 22), (6, 22), (7, 22), (8, 22), (9, 22), (5, 13), (6, 13), (7, 13), (5, 21), (6, 21) }  
Best cost so far: 158  
Tabu list used?: Yes  
Selected neighbour: [29, 3, 2, 19, 18, 22, 9, 8, 13, 7, 21, 6, 5, 20, 17, 16, 15, 12, 28, 27, 11, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]  
Selected neighbour cost: 158

**Results** The best schedule  $x_{TS}$  found was:

[29, 3, 2, 19, 18, 22, 9, 8, 13, 7, 21, 6, 5, 20, 17, 16, 15, 12, 28, 27, 11, 26, 25, 24, 23, 4, 1, 14, 10, 0, 30]

Which had a total tardiness of **158**. Here it seems that the algorithm did effectively converge but the tabu list was not large enough to prevent cycling, as the best solution was found more than once. The algorithm could be improved by increasing the tabu list length to prevent revisiting recent states and cycling.

## 2.2 Parameter Exploration and Results

The best schedule  $x_{TS}$  found was:

[29, 3, 2, 9, 8, 7, 6, 22, 19, 5, 18, 21, 13, 20, 17, 16, 15, 12, 28, 27, 26, 25, 24, 23, 11, 4, 1, 14, 10, 0, 30]

with a total tardiness of **158**.

### Parameters Used

- **Tabu List Length ( $L$ ):** 60 - this tabu list length was set to 60 to allow sufficient history to avoid cycling while maintaining exploration flexibility in the neighborhood (up to 29 adjacent swaps)
- **Maximum Iterations ( $K$ ):** 500 - this value was chosen to ensure the algorithm had enough time to converge to a good solution
- **Threshold for Accepting Worse Solutions ( $\gamma$ ):** 20 - this value balanced allowing worse solutions to escape local optima and converging efficiently

## Parameter Effects

### 1. Tabu List Length ( $L$ ):

- For smaller values of  $L$  (e.g.,  $L = 20$ ), the tabu list wasn't large enough to prevent revisiting recent states, leading to slower improvement, cycling and suboptimal solutions
- Increasing  $L$  to 60 was good for avoiding cycles and also allowing sufficient exploration of the neighborhood
- Larger values of  $L$  (e.g.,  $L = 80$ ) restricted the search too much and occasionally skipped good moves, resulting in worse solutions

### 2. Maximum Iterations ( $K$ ):

- Setting  $K$  to 500 allowed the algorithm to converge fully, in fact it only needed 374 iterations
- Reducing  $K$  (e.g.,  $K = 200$ ) often resulted in suboptimal solutions, as the search terminated prematurely before the algorithm could escape local optima
- Increasing  $K$  beyond 500 did not result in significant improvements but increased runtime unnecessarily, and for cases where the tabu list was small, it just cycled

### 3. Threshold for Accepting Worse Solutions ( $\gamma$ ):

- Smaller values of  $\gamma$  (e.g.,  $\gamma = 10$ ) caused the algorithm to become stuck in local optima, as fewer worse solutions were accepted, reducing exploration
- Larger values of  $\gamma$  (e.g.,  $\gamma = 30$ ) allowed too much exploration, which slowed convergence and resulted in noisy search behavior, but did sometimes find better solutions
- $\gamma = 20$  provided a good tradeoff between exploration and exploitation, letting the algorithm escape local optima while still converging efficiently

## Conclusion

The best schedule was found with  $L = 60$ ,  $K = 500$ , and  $\gamma = 20$ , achieving a total tardiness of **158**. These parameters provided a good balance between avoiding cycling, allowing sufficient exploration, and converging efficiently. Parameter tuning revealed that values for  $L$  and  $\gamma$  which were too small or too large degraded performance, and  $K$  needed to be just large enough to allow full convergence without unnecessary runtime overhead.