Faculty of Engineering and Technology
Electrical and Computer Engineering Department

ENCS3340-ARTIFICIAL INTELLIGENCE

Project#1

Prepared by:

Aws Shaheen	**1212585**	**sec. 1**
Alhasan Manasrah	**1211705**	**sec. 3**

Instructors: Yazan Abu Farha, Aziz Qaroush

Deadline: 20-5-2024

Test case for three jobs

```
Enter the number of jobs: 3

Entering details for Job 1:
Enter machine ID (e.g., M1) or 'done' to finish: M1
Enter processing time for M1: 12
Enter machine ID (e.g., M1) or 'done' to finish: M2
Enter processing time for M2: 5
Enter machine ID (e.g., M1) or 'done' to finish: M3
Enter processing time for M3: 7
Enter machine ID (e.g., M1) or 'done' to finish: M4
Enter processing time for M4: 4
Enter machine ID (e.g., M1) or 'done' to finish: done

Entering details for Job 2:
Enter machine ID (e.g., M1) or 'done' to finish: M1
Enter processing time for M1: 6
Enter machine ID (e.g., M1) or 'done' to finish: M3
Enter processing time for M3: 11
Enter machine ID (e.g., M1) or 'done' to finish: M4
Enter processing time for M4: 9
Enter machine ID (e.g., M1) or 'done' to finish: done

Entering details for Job 3:
Enter machine ID (e.g., M1) or 'done' to finish: M1
Enter processing time for M1: 5
Enter machine ID (e.g., M1) or 'done' to finish: M2
Enter processing time for M2: 12
Enter machine ID (e.g., M1) or 'done' to finish: M4
Enter processing time for M4: 8
Enter machine ID (e.g., M1) or 'done' to finish: done
```

Here the program asks the user to enter the number of jobs and the details for each job, like machine ID, and the processing time for each machine.

```
Jobs and their operations have been entered successfully.
Job_1: M1[12] -> M2[5] -> M3[7] -> M4[4]
Job_2: M1[6] -> M3[11] -> M4[9]
Job_3: M1[5] -> M2[12] -> M4[8]

Machine schedules:
M1: 12, 6, 5
M2: 5, 0, 12
M3: 7, 11, 0
M4: 4, 9, 8

Best Fitness (makespan): 28

Detailed Scheduling Info:
Machine M1:
  Job 2: Start at 0, End at 6
  Job 1: Start at 11, End at 23
  Job 3: Start at 23, End at 28
Machine M2:
  Job 3: Start at 0, End at 12
  Job 1: Start at 23, End at 28
Machine M3:
  Job 1: Start at 4, End at 11
  Job 2: Start at 15, End at 26
Machine M4:
  Job 1: Start at 0, End at 4
  Job 2: Start at 6, End at 15
  Job 3: Start at 15, End at 23

Gantt Chart:
Machine 1: 222222     11111111111133333
Machine 2: 333333333333           11111
Machine 3:     1111111    22222222222
Machine 4: 1111  22222222233333333
```

After entering the details above, the program displays all jobs and work on which machines, then Each machine times for every job, scheduling info, and Gantt chart below.

Problem formulation

In this test case the problem involves three jobs with sequence of operations that need to be processed on various machines. The objective is to minimize the total time required to complete all jobs. The operations for each job are as follows:

Job 1: M1[12] -> M2[5] -> M3[7] -> M4[4]

Job 2: M1[6] -> M3[11] -> M4[9]

Job 3: M1[5] -> M2[12] -> M4[8]

Chromosome Representation

Each chromosome represents a list of operations where each operation is represented by a tuple containing job number, Machine ID, start time, duration like this:

[(1, 'M1', 0, 12), (1, 'M2', 12, 5), (1, 'M3', 17, 7), (1, 'M4', 24, 4), (2, 'M1', 0, 6), (2, 'M3', 6, 11),

(2, 'M4', 17, 9), (3, 'M1', 0, 5), (3, 'M2', 5, 12), (3, 'M4', 17, 8)].

Cross over

The crossover operation combines parts of two parent chromosomes to produce children,
a crossover point is selected such that it does not split operations of the same job,
the child is a representation of initial segment from one parent and the remaining operations from the other parent, ensuring no duplicate operations, this is an example when the crossover point is after the operations of Job 1:

parent1

[(1, 'M1', 0, 12), (1, 'M2', 12, 5), (1, 'M3', 17, 7), (1, 'M4', 24, 4), (2, 'M1', 0, 6),

(2, 'M3', 6, 11), (2, 'M4', 17, 9), (3, 'M1', 0, 5), (3, 'M2', 5, 12), (3, 'M4', 17, 8)]

Parent2

[(2, 'M1', 0, 6), (2, 'M3', 6, 11), (2, 'M4', 17, 9), (1, 'M1', 0, 12), (1, 'M2', 12, 5),

(1, 'M3', 17, 7), (1, 'M4', 24, 4), (3, 'M1', 0, 5), (3, 'M2', 5, 12), (3, 'M4', 17, 8)]

After cross over:

Child1

[(1, 'M1', 0, 12), (1, 'M2', 12, 5), (1, 'M3', 17, 7), (1, 'M4', 24, 4), (2, 'M1', 0, 6),

(2, 'M3', 6, 11), (2, 'M4', 17, 9), (3, 'M1', 0, 5), (3, 'M2', 5, 12), (3, 'M4', 17, 8)]

Child2

[(2, 'M1', 0, 6), (2, 'M3', 6, 11), (2, 'M4', 17, 9), (1, 'M1', 0, 12), (1, 'M2', 12, 5),

(1, 'M3', 17, 7), (1, 'M4', 24, 4), (3, 'M1', 0, 5), (3, 'M2', 5, 12), (3, 'M4', 17, 8)]

Mutation

The mutation is swapping two operations within the same job with a small probability.

Original Chromosome

[(1, 'M1', 0, 12), (1, 'M2', 12, 5), (1, 'M3', 17, 7), (1, 'M4', 24, 4), (2, 'M1', 0, 6),

(2, 'M3', 6, 11), (2, 'M4', 17, 9), (3, 'M1', 0, 5), (3, 'M2', 5, 12), (3, 'M4', 17, 8)]

Mutated Chromosome

[(1, 'M2', 12, 5), (1, 'M1', 0, 12), (1, 'M3', 17, 7), (1, 'M4', 24, 4), (2, 'M1', 0, 6),
(2, 'M3', 6, 11), (2, 'M4', 17, 9), (3, 'M1', 0, 5), (3, 'M2', 5, 12), (3, 'M4', 17, 8)]

Objective Function

The objective function evaluates the fitness of each chromosome by calculating the makespan. The makespan is the maximum end time across all machines, representing the total time required to complete all jobs. The goal is to minimize this makespan.

In the provided test case the makespan is 28