ACP Summer School 2026

CP: Explanation, Verification and Applications

2026年吉林大学计算机科学拔尖人才培养基地“成长伙伴”国际暑假学校

Optimisation Challenge: Airport Turnaround Scheduling

Changchun, China
24 – 28 August 2026



Why Hackathon?

- Learn swimming in the swimming pool
- Tackle a real-ish optimisation problem
- Meet fellow optimisers and make friends
- Choose any solving technology you like



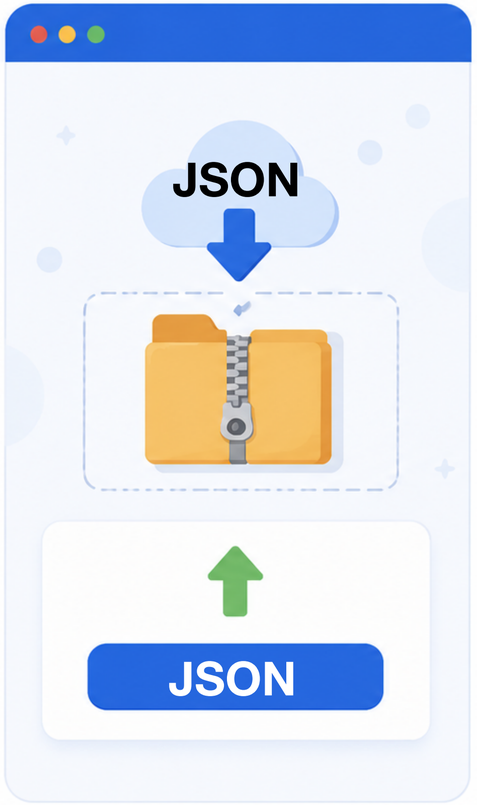
Turnaround Scheduling

- Daily operations for Mel-Airport
  - 500+ aircraft land and depart
  - Every one needs a parking lot
  - Grounds crew to turn it around
- You are the operation planner:
  - **Which gate** each flight parks at
  - **When** each ground task runs
  - **Which team** performs which task

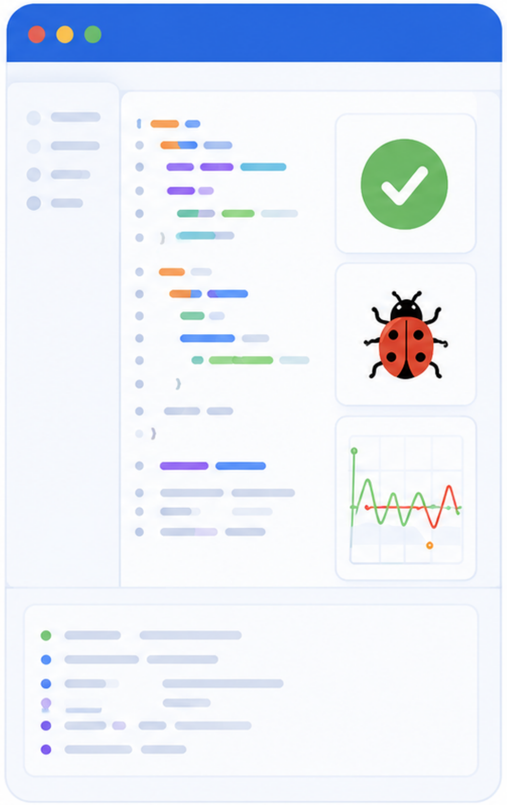


Hackathon: Optimisation Challenge

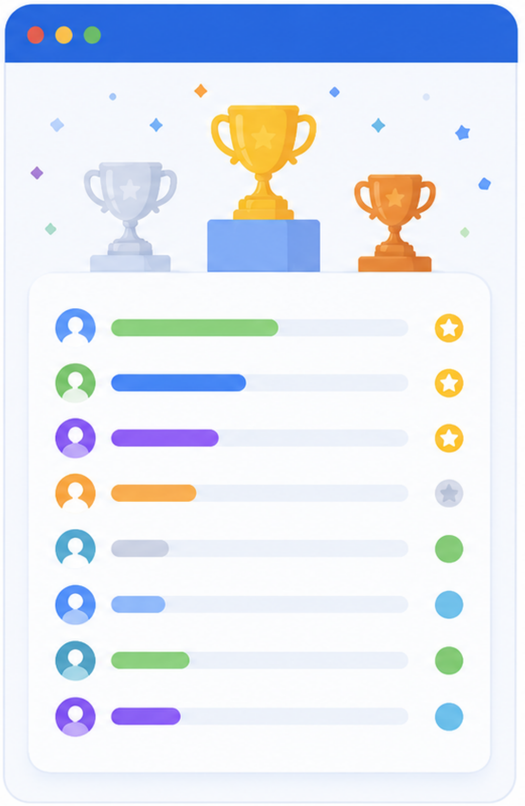
- Website: https://hackathon.minizinc.dev/



Submission



Debug



Leaderboard

Hackathon: Submission

ACP Summer School 2026 Hackathon Submission Portal

Submit | Leaderboard | Debug

Submissions close in **3d 04:43:14** | Deadline **28 Aug 2026, 16:00** | Leaderboard freeze **28 Aug 2026, 15:30**

Get the bundle

Everything you need to start solving the problem.

- hackathon.pdf — problem specification
- data — folder with instance files
- Project.mzp — a MiniZinc project file
- airport_sched.mzn — starter MiniZinc model
- starter.py — starter Python script

Download bundle (.zip)

Submit solutions

Drop one or many `sol_XX.json` files. We infer the instance from the filename and grade each one independently. No team token yet? Sign up here, we'll email you a token.

Team token

team-...

Emailed to your team after signing up via the form.

Solution files

Drop `sol_XX.json` files here, or click to browse

Multiple files OK. Instance ID auto-detected from filename (e.g. `sol_06.json` → hackathon_06)

Grade 0 file(s)

Hackathon: Debug

ACP Summer School 2026 — Debug Viewer

Drop your solution json file (name must be `sol_XX.json` or `sol_XX_<slug>.json`) and see gate/team/flight views + violations. Nothing submitted — this is for your eyes only.

← Submit Leaderboard

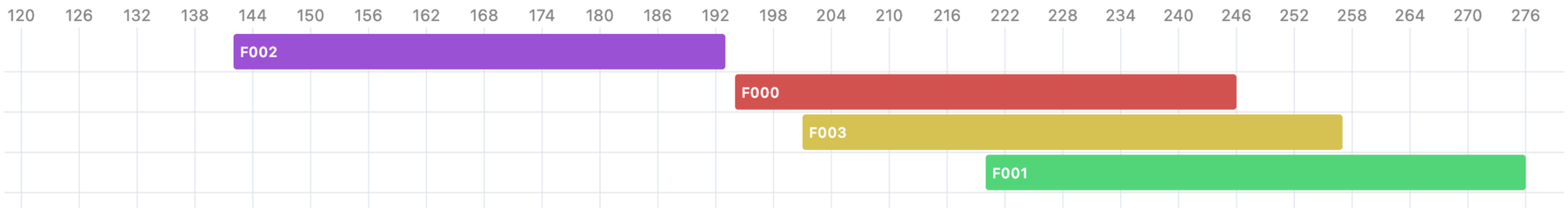
Solution: Choose file sol_01.json Instance auto-detected from filename (e.g. `sol_02_*.json` → hackathon_02). Instance: 4 flights, 4 gates, 57 teams Cost: 2522

T1 C1-C5 ok **T2** + C6-C10 ok **T3** + C11-C12 ok

Gate view Team view Flight view

4 flights, horizon 280 slots | 0 violation(s). Click a bar to spotlight that flight across all views. Click a violation in the sidebar to flash the bars involved.

Flight (colour per flight); border red = violation.



Gate violations (0)

No violations in this view.

Hackathon: Leaderboard

Submissions close in **15:43:17** | Deadline **29 Jul 2026, 15:30** | Freeze **29 Jul 2026, 15:00**

Standings — points per instance (higher is better, out of 100), total out of 1000

Updated 23:46:36 · 4 teams

#	TEAM	01	02	03	04	05	06	07	08	09	10	TOTAL
1	**The Flying Kangaroo** @monash	100.0 T3	20.0 T1	—	100.0 T3	—	91.6 T3	100.0 T3	100.0 T3	100.0 T3	—	**611.6**
2	**Melbourne Run!** @melbourne	—	100.0 T3	—	89.9 T3	—	97.7 T3	96.9 T3	97.5 T3	—	—	**482.0**
3	**A Good Day** @rmit	—	20.0 T1	100.0 T3	—	100.0 T3	97.7 T3	—	97.5 T3	—	—	**415.2**
4	**Crazyflight** @mel-uni	—	20.0 T1	100.0 T3	89.9 T3	—	97.7 T3	—	—	—	—	**307.6**

Highest points on this instance | Tier 3 (correct) | Tier 2 (violates C10/C11) | Tier 1 (gate only) | — = no submission

Hackathon: Instance Points

- The points for each of 12 instances is calculated individually:

	Constraints	Problem Scope	Points formula
Tier 1	C1 to C5	Gate assignment	20
Tier 2	C1 to C10	+ Tasks assignment and scheduling	+ 50 x (best cost / your cost)
Tier 3	C1 to C12	+ Travel and rest constraints	+ 30 x (best cost / your cost)

- Instance points = Tier 1 points + Tier 2 points + Tier 3 points

Let’s Get Start!

- Find your teammates and sign up!
- Download the bundle in the website: https://hackathon.minizinc.dev/
- Work on gate assignment solutions!

Good Luck!
The apron is yours!

Today: Gate Assignment

- Flights have attributes:
  - Arrival time and departure times
  - Wide/narrow-body, passenger/freighter, international/domestic flights
- Gates have attributes:
  - Open time and close times
  - Large/small stand, passenger/freighter, international/domestic gates
- Handle uncertainty: 30-minute between flights at the same gate
- Hint: think about the “alldifferent” global constraint

Today: Data Format

Listing 1: Instance JSON

```
{
  "horizon":    1440,
  "FlightID":   ["JST917", "UAE9314", "JST034", "MXD177", "CPA3128"],
  "GateID":     ["D02", "D03", "D04", "H03"],
  "LaborID":    ["ramp_team_01", "ramp_team_02", "baggage_team_01",
                 /* ... one per team unit ... */],
  "max_tasks": 11,

  /* Data for flights */
  "flight_arr": [31, 754, 242, 256, 855],
  "flight_dep": [76, 889, 287, 301, 990],
  "flight_carrier": ["domestic","international","domestic",
                     "domestic","domestic"],
  "flight_size": ["narrow","wide","narrow","narrow","wide"],
  "flight_op_type": ["passenger","passenger","passenger",
                     "passenger","freighter"],
```

The number of minutes

Today: Data Format

Listing 1: Instance JSON

Array indexed by FlightID

```
{
  "horizon":    1440,
  "FlightID":   ["JST917", "UAE9314", "JST034", "MXD177", "CPA3128"],
  "GateID":     ["D02", "D03", "D04", "H03"],
  "LaborID":    ["ramp_team_01", "ramp_team_02", "baggage_team_01",
                 /* ... one per team unit ... */],
  "max_tasks": 11,

  /* Data for flights */
  "flight_arr": [31, 754, 242, 256, 855],
  "flight_dep": [76, 889, 287, 301, 990],
  "flight_carrier": ["domestic","international","domestic",
                     "domestic","domestic"],
  "flight_size": ["narrow","wide","narrow","narrow","wide"],
  "flight_op_type": ["passenger","passenger","passenger",
                     "passenger","freighter"],
```

Today: Data Format

Listing 1: Instance JSON

```
{
  "horizon":    1440,
  "FlightID":   ["JST917", "UAE9314", "JST034", "MXD177", "CPA3128"],
  "GateID":     ["D02", "D03", "D04", "H03"],
  "LaborID":    ["ramp_team_01", "ramp_team_02", "baggage_team_01",
                 /* ... one per team unit ... */],
  "max_tasks": 11,

  /* Data for flights */
  "flight_arr": [31, 754, 242, 256, 855],
  "flight_dep": [76, 889, 287, 301, 990],
  "flight_carrier": ["domestic","international","domestic",
                     "domestic","domestic"],
  "flight_size": ["narrow","wide","narrow","narrow","wide"],
  "flight_op_type": ["passenger","passenger","passenger",
                     "passenger","freighter"],
```

Flight JST917 lands at 00:31 and departs at 01:16

Today: Data Format

Listing 1: Instance JSON

```
{
  "horizon":    1440,
  "FlightID":   ["JST917", "UAE9314", "JST034", "MXD177", "CPA3128"],
  "GateID":     ["D02", "D03", "D04", "H03"],
  "LaborID":    ["ramp_team_01", "ramp_team_02", "baggage_team_01",
                 /* ... one per team unit ... */],
```

Array indexed by GateID

```
/* Data for gates */
"gate_stand_size": ["small","large","large","large"],
"gate_usage":      ["domestic","international","international","domestic"],
"gate_op_type":    ["passenger","passenger", "passenger","freighter"],
"gate_open":       [0, 0, 0, 360],
"gate_close":      [1440, 1440, 1440, 1440],
```