

Assignment 5

For this assignment, form groups of two. This assignment is worth a total of 40 points. Solutions must be submitted by **31 May 2026**. The assignment presentations will take place on **1 June 2026**.

Use the link on e-učilnica to submit your work. The report must be in `.pdf` format. You should also submit the source code of your implementation and the results in `.txt` format.

Continuous optimization of CEC 2022 functions

The aim of this assignment is to find the best result for 12 optimization (minimization) functions that are available in the `opfunu` package [2]. The functions used will be from the CEC 2022 benchmark functions [1].

Instructions

To test the results, use all 12 functions with 20 dimensions and the default bounds, which are in the range from -100 to $+100$ for each dimension. You can see the lower and upper bounds of each function in the variables `lb` and `ub`. See the appendix and supplementary code for an example.

As a team, you need to implement at least two optimization programs. One of the optimization programs must be a local search method, such as best descent local search, tabu search, guided local search, variable neighborhood search, simulated annealing, or another related method. The second one can be any optimization approach of your choice, such as a genetic algorithm, differential evolution, whale optimization algorithm, ant colony optimization, or another approach. The second approach can also be a local search method.

You need to implement the approaches yourselves and must not use already built optimizers from other packages. There is no time limit on how long you can let the algorithms run.

Reporting

Write your results into the Google Spreadsheet available here:

https://docs.google.com/spreadsheets/d/10nRzZSaDyLOmuMS6WaQa-0QIqy_vhLJfxsLOXcwbxic/edit?usp=sharing

Each group must write its group information in the spreadsheet no later than **25 May 2026**, one week before the presentations. This deadline is only for group formation.

Each group is encouraged to fill in its results as soon as it has them available, so that peers can see the currently best obtained results. The final results should be written in the spreadsheet no later than the assignment submission deadline, **31 May 2026**.

Write a report with your results and a description of the approaches used. The report should include the results for each tested method separately and a short description of each method, approximately 1–2 pages per method.

Submit your code and report on e-učilnica.

You should also submit the coordinates of the found minimums for each method in a `.txt` file. The file should contain 12 lines, each line representing one function. The coordinate values should be separated by tabulators. See the example on e-učilnica.

Grading

The final grade will be based on:

- quality of results,
- quality of the report,
- oral presentation,
- number of methods tested,
- code quality.

Appendix

Python example

```
from opfunu.cec_based import cec2022

# Choose the CEC 2022 benchmark functions
functions = [
    cec2022.F12022, cec2022.F22022, cec2022.F32022,
    cec2022.F42022, cec2022.F52022, cec2022.F62022,
    cec2022.F72022, cec2022.F82022, cec2022.F92022,
    cec2022.F102022, cec2022.F112022, cec2022.F122022
]

for f in functions:
    try:
        # Initialize the 20D function
        func = f(ndim=20)
```

```

# Find the recommended bounds and evaluate function at point 0
lower = func.lb
upper = func.ub
x = [0] * 20

# Evaluate the function
fitness = func.evaluate(x)

# Print results
print(f"{f.__name__}: {fitness}={fitness:.3f}")

except Exception as e:
    print(f"Error: {f.__name__}: {e}")

```

References

- [1] Opfunu CEC 2022 based module.
Opfunu documentation: CEC 2022 based module.
Accessed: 2026-05-05.
- [2] Nguyen Van Thieu. *Opfunu: an open-source Python library for optimization benchmark functions*. Journal of Open Research Software, 12(1), 2024.