



#### НАПОМЕНЕ

1. Обавезно прочитати **СВЕ** напомене.
2. Решења писати у оквиру ручно направљеног *cargo* пројекта.
3. Обавезно уписати име, презиме и број индекса у коментар на почетку датотеке која представља *entrypoint* програма.
4. Решење које не може да се компајлира носи **0 поена**.
5. Коментарисати оне делове за које сматрате да би коментари додатно појаснили комплексну логику.

## Спецификација пројектног задатка: Тура кроз европске градове

### Опис пројекта

Студент је ангажован за развој програма који планира туру кроз групу европских градова, водећи рачуна о укупној пређеној удаљености. Тура, која представља резултат рада алгоритма, треба да обухвати све градове из задатог списка тако да се свака посети тачно једном, а на крају се пут завршава повратком у полазни град. Циљ алгоритма је да одреди што краћу могућу туру.

Програм треба да буде написан у програмском језику *Rust* и да обухвати две имплементације:

- **Секвенцијална** - Једнонитна верзија.
- **Паралелна** - Вишенитна верзија, потенцијално убрзана.

**Улазни подаци:** Списак градова и растојања између истих (у километрима) налазе се у *.txt* датотеци која се налази као прилог овој спецификацији.

### Оквирне смернице

- Развити секвенцијалну верзију за планирање туре кроз дат број градова.
- Паралелизовати програм како би се искористила сва расположива језгра процесора.
- Упоредити перформансе секвенцијалне и паралелне верзије.
- Припремити извештај који објашњава коришћени приступ и резултате мерења.

### Технологије и алатке

Обавезно је користити следеће технологије и алате:

- **Програмски језик:** *Rust* (мин. верзија 1.75, најсвежија је свакако добродошла).
- **Развојно окружење:** *cargo* (мин. верзија 1.75, најсвежија је свакако добродошла).



Опционално је (али се препоручује) користити следеће технологије и алате:

- **Rust библиотеке:** За паралелизам и рад са матрицама.
- **Python:** За визуализацију резултата.

## Приступ и алгоритми

Студент има слободу да одабере један или више приступа за планирање тура. Решење не мора бити строго оптимално; важније је демонстрирати рад изабраног алгоритма и мерити његову ефикасност. У извештају објаснити разлоге за избор метода и евентуално комбиновање више техника.

## Извештај и презентација резултата

Неопходно је саставити извештај који у себи садржи следеће целине.

- Кратак опис изабраног алгоритма и мотивације за избор.
- Метрика перформанси: Измерити времена извршавања секвенцијалне и паралелне верзије за најмање три различита броја градова (нпр.  $n = 8, 12$  и  $16$ ).
- Графички и/или табеларни приказ.
- Израчунавање убрзања и дискусија о уским грлима.