

Задатак 1. Рођенданска журка и друшћивене иџре

Јована организује рођенданску журку. Како би анимирала своје госте, припремила је низ друштвених игара. Једна од њих је игра „Разоружавање земаља“. Марији се свиђа игра, али зна да играње такве игре дуго траје јер има много комбинација које треба пробати. Ипак Марија је у журби, она жели да што пре дође до решења како би стигла кући раније и учила свој омиљени предмет **Увод у вештачку интелигенцију**. Зато јој је потребна Ваша помоћ око брзог проналаска решења. Размотримо игру у наставку:

Земље X и Y преговарају о разоружању. Свака има пројектиле различите снаге, при чему је снага изражена једним позитивним целим бројем. Сваки корак у преговорима резултује једном од следеће две акције:

1. елиминише са обе стране по један пројектил исте снаге, или
2. елиминише на једној страни пројектил снаге S , а на другој два пројектила чија је укупна снага S .

Навести низ корака који ће довести до елиминације свих пројектила са једне или обе стране, при чему ни у једном кораку ниједна страна не сме имати више од једног пројектила од друге стране.

Пример почетка игре:

X	9	11	9	3	4	5	7	18
Y	12	2	16	5	7	4	20	

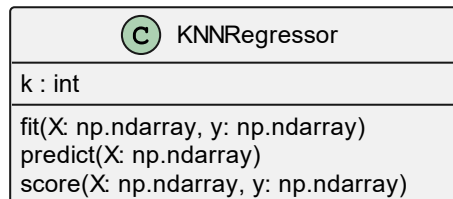
Написати класу *Problem* која дефинише начин записивања стања, функцију транзиције и проверу испуњења циља. Изабрати што погоднији алгоритам претраживања како би Марија што брже дошла до решења и исписати одигране потезе. Дати кратак коментар о разлогу избора алгоритма.

Задатак 2. Цене дијаманата

На основу историјских података продаји дијаманата креирати моделе машинског учења којима се може проценити цена дијаманата. Сет податка `train.csv` намењен је обуци модела и њиховим оценама.

Потребно је извршити следеће кораке:

1. Учитати сет податка `train.csv`.
2. Проверити да ли неке вредности података недостају. Уколико недостаје део података, попунити просечним вредностима за тај тип података уколико се ради о нумеричким вредностима, односно најчешћом категоријом уколико се ради о категоријским варијаблама.
3. Све категоријске варијабле кодирати методом *LabelEncoding* (постоји више начина за реализацију – сви су прихватљиви).
4. Поделити податке на скуп за обуку и скуп за валидацију у односу 60:40
5. Потребно је имплементирати класу *KNNRegressor* која извршава алгоритам К најближих суседа. У наставку дат је UML дијаграм класе *KNNRegressor*.



- a. За дистанцу користити Еуклидско растојање
 - b. Функција за оцену модела може користити r^2 метрику
6. Употребом претходно креиране класе *KNNRegressor* обучити моделе за различите вредности k , $1 \leq k \leq 20$.
7. На истом дијаграму исцртати оцене креираних модела на тренинг и валидационом скупу у зависности од параметра k .
8. На основу креираних експеримената одредити најпогоднију вредност параметра k у циљу добијања што генералнијег модела.
9. За одабрано k обучити модел типа *sklearn.neighbors.KNeighborsRegressor*.
10. Оценити модел из тачке 9. на тренинг и валидационом скупу.

Могуће је креирати и додатне кораке које сматрате релевантним и неопходним за исправно понашање модела машинског учења.

НАПОМЕНА: Израда колоквијума траје 120 минута. Свој рад предати у облику *Python* скрипте или *jupyter notebook*-а и сместити у директоријум `Desktop\Rad\indeks_ime_prezime`.