Laborator 2

Saptamana 2-6 martie 2020

Continut:

1. Setup
2. Exercitii cu functii
3. Exercitii cu NumPy

Setup

Formati ecipe de cate doi. Creati un repository *privat* pe github, cu numele: IDS_NumeFamilieStudent1_NumeFamilieStudent2. Indiferent de grupa din care faceti parte, adaugati ca si colaboratori pe acest repo utilizatorii: akerestely si lmsasu.

Resurse:

1. Setare de repository privat (https://help.github.com/en/github/administering-a-repository/setting-repository-visibility)
2. Clienti de git (https://www.hostinger.com/tutorials/best-git-gui-clients/)

Exercitii cu functii

Nota: aceste exercitii nu se puncteaza.

1. Sa se scrie o functie care primeste ca parametru o lista de numere si returneaza un tuplu continand: suma elementelor si diferenta maxima dintre elementele listei.
2. Scrieti o functie care preia doua liste si returneaza `True` daca cele doua liste de numere contin cel putin k elemente comune ($k \geq 1$ dat ca parametru intreg, valoare implicita 1), `False` altfel.
3. Sa se scrie o functie care determina daca o lista contine cuvinte reprezentand doar numere intregi (optional: numere fractionare, scrise sub forma parte intreaga, optional urmata de punct zecimal, optional urmata de parte fractionara: 1, 2., 13.14) (Puteti folosi modulul `re` pentru lucrul cu expresii regulate.) Functia trebuie sa testeze daca:
   A. elementele din lista sunt stringuri
   B. fiecare din stringuri este un numar
4. Sa se scrie o functie care returneaza numarul de vocale si de consoane dintr-un parametru dat - rezultatul returnat de functie este tuplu cu doua valori. Cuvantul se va converti la litere mici in interiorul functiei.
5. Sa se scrie o functie care primind o lista de dictionare, returneaza `True` daca toate dictionarele contin cel putin n elemente, n transmis ca parametru, `False` altfel.
6. Sa se scrie o functie recursiva care sa faca ridicarea la putere a unui numar, astfel:
$$a^n = \begin{cases} a & \text{daca } n = 1 \\ \left(a^{n//2}\right)^2 & \text{daca } n \text{ e par} \\ \left(a^{n//2}\right)^2 \cdot a & \text{daca } n \text{ e impar}, n > 1 \end{cases}$$
unde prin $n//2$ se reprezinta catul impartirii intregi intre n si 2.

Exercitii simple folosind NumPy

Nota: in rezolvarea exercitiilor se vor folosi cat mai mult functii NumPy *vectorizate*. Functiile scrise de voi vor fi documentate cu docstrings (tutoriale: Docstrings in Python, urmati NumPy Style Python Docstrings). Folositi type annotations. Utilizati doar pachetele NumPy.

1. (0.5 puncte) Pentru un vector NumPy dat, sa se determine care sunt elementele unice, in ordine descrescatoare. Exemplu: pentru `x = np.array([-4, 3, 5, 3, 2, 1, -3, -4, 2, -4])` functia scrisa de dvs va intoarce rezultatul `[ 5, 3, 2, 1, -3, -4]`.
2. (0.5 puncte) Pentru un vector NumPy dat, sa se determine care e elementul cel mai frecvent si cel mai putin frecvent (tuplu de 2 valori). Daca exista mai multe numere cele mai frecvente (cele mai putin frecvente), se va raporta oricare din ele. Exemplu: pentru `x = np.array([-4, 3, 5, 3, 2, 1, -3, -4, 2, -4])` functia scrisa de dvs va intoarce rezultatul `(-4, -3)`.
3. (0.5 puncte) Plecand de la o matrice, sa se scrie o functie care returneaza vectorul minimelor de pe linii si vectorul maximelor de pe coloane. Exemplu: pentru matricea `mat = np.array([[1, 2, 3, 4], [1, -2, 3, -4], [3, 10, 3, 4]])` functia va returna tuplul de vectori: `(array([ 1, -4, 3]), array([ 3, 10, 3, 4]))`. Indicatie: folositi-va de parametrul `axis`.
4. (0.5 puncte) Pentru o matrice de numere, sa se determine produsul elementelor pe linii sau de pe coloane, in functie de valoarea unui parametru boolean `compute_on_lines`. Exemplu: pentru matricea `mat = np.array([[1, 2, 3, 4], [1, -2, 3, -4], [3, 10, 3, 4]])` si parametrul `compute_on_lines=True`, functia va returna vectorul `[ 24 24 360]`, iar pentru `compute_on_lines=False` se va returna `[ 3 -40 27 -64]`.
5. (0.5 puncte) Determinati daca macar unul din elementele unei matrice `a` este mai mare decat cel de pe pozitie corespondenta din `b` (adica daca avem macar o pereche de indecsi (i, j) pentru care a[i, j] > b[i, j]). Daca da, se va returna o astfel de pereche de indecsi; daca nu, se va returna None.
6. (0.5 puncte) Construiti o functie care, primind o matrice, determina pe ce pozitii se afla valorile in afara unui interval [min, max] dat prin parametri.
7. (1 punct) Se considera vectorul: `a = np.array([230, 10, 284, 39, 76])`. Folosind indexarea logica, sa se inmulteasca elementele din vector care sunt mai mici ca 100 cu 2 (restul raman neschimbate). Folosind ciclare, sa se aplice aceeasi operatie pana cand toate elementele devin mai mari ca 100. Sa se afiseze din vectorul final elementele care sunt mai mari de 150 si mai mici de 200.
8. (1 punct) Sa se scrie o functie `pair_max` care preia doi vectori de aceeasi lungime si returneaza maximele pe pozitiile corespunzatoare: `python`
   ```
   a = np.array([3, 7, 9, 13, -10, 200, 3])
   b = np.array([4, 5, -9, 100, 300, 230, 1])
   pair_max(a, b)
   #iesire dorita:
   array([ 4,  7,  9, 100, 300, 230,  3])
   ```
   . Functia va verifica cu `assert` ca vectorii au lungimi egale.
9. (1 punct) Descarcati setul de date Iris de la adresa '[https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data (https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)](https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)', si incarcati-l intr--o matrice NumPy, folosind `np.genfromtxt`. Alegeti aleator 10 pozitii in matricea de 150 linii si 4 coloane (omiteti ultima coloana, de tip text), setati aceste valori pe NaN. Construiti o functie care, primind la intrare o matrice, returneaza un tuplu cu indicii de linii respectiv de coloane in care se gasesc valori NaN.
10. (1 punct) Sa se construiasca o functie care returneaza cele mai mici `k` valori dintr-un vector de cel putin `k` elemente, impreuna cu pozitiile lor.

Precizari

1. Se acorda 3 puncte din oficiu.
2. Tema se va prezenta in saptamana 16-20 martie 2020.
3. Repository-ul va contine un fisier "README.md" (in radacina repository-ului) in care vor fi mentionati autorii temelor, cu nume complet si grupa din care fac parte.