Introduction to Computer Programming

Exercises – Week 9. NumPy

Exercise 1 - Creating arrays and accessing elements (Essential)

1. Create a variable a that stores the array [5, 4, 9, 2, 0, 4, 7, 2].
2. Print the first entry of a and the last entry of a (**Hint**: You can use the index -1 to access the last entry of an array). The colon operator : can be used to access sequential elements of an array. Print the values of `a[1:6]` and explain the output.
3. Change the last entry of a to -9 and print the result. Now run the command `a[0:3] = 1` and print the result. How has this altered the array a?
4. Create an array r that contains 20 random integers between 1 and 9 (inclusive). Print the result. This array will be used in the next question.
5. (Advanced) **Logical indexing** provides a quick way to access and modify entries in an array that satisfy certain criteria. In this question, we'll use logical indexing to replace all of the entries in r that are smaller than 5 with 0. First, run the command `idx = r < 5`. Print the value of `idx`. Explain the result you see. Now run the command `r[idx] = 0` and print the value of r. What has happened?
6. Create a variable A to store the matrix

$$\begin{pmatrix} 6 & 2 & 3 \\ 4 & 4 & 1 \\ 8 & 5 & 6 \end{pmatrix} \tag{1}$$

   as a NumPy array.
7. Change the entry in the second row, first column of A to 9. Then change the entry in the last row and last column of A to 0. Print the updated array A. The n-th row of A can be accessed using the colon operator as `A[n-1,:]`. Similarly, the m-th column of A can be accessed using `A[:,m-1]`. Use the colon operator to print the entries in the second row of A.
8. Create a 2×2 array of zeros and assign it to a variable A. Use the colon operator to set the first row of A to 1 and the second row of A to 2. **Hint**: The operation `A[n-1,:] = q` will set all of the entries in the n-th row of A to the value q. Using a `for` loop, create a 5×5 matrix where the entries in row i are equal to i.

Exercise 2 - Performing operations on NumPy arrays (Essential)

1. Create two NumPy arrays to store the vectors $a = (3, 5, 2)$ and $b = (6, 3, 1)$. Calculate $c = a + 2b$. Calculate the dot product $a \cdot b$ using the `dot` method or the `dot` function. Can you also compute the dot product using element-by-element multiplication along with the `np.sum` function? Recall that the dot product is defined as $a \cdot b = \sum_i a_i b_i$.
2. Create an array called t that contains 500 values between 0 and 5. Create a second array called y that stores the values of $y = t^2 e^{-2t}$. **Hint**: use the `exp` function to compute the exponential of a NumPy array. Find the maximum value of y. **Note**: this is a simple way of

finding the maximum of a function. (Advanced): Use logical indexing or otherwise to find the value of t at which y is maximal.

3. This question will demonstrate that NumPy can be used to integrate functions. Create a NumPy array x that stores 50 values between 0 and 5. Create the array $y = x/(x+1)$. Look up how to use NumPy's `trapz` function, which uses the trapezoid rule to approximate integrals. Use `trapz` to compute $I = \int_0^5 y(x)\,dx$. The exact value is $I = 5 - \ln(6) = 3.208240530...$ What happens if you repeat the calculation using 500 points between 0 and 5?
4. The table below provides the gravitational acceleration g of each of the planets:

Planet	g [m/s^2]
Mercury	3.7
Venus	8.9
Earth	9.8
Mars	3.7
Jupiter	25
Saturn	10
Uranus	8.9
Neptune	11

   Use NumPy functions to compute the maximum, minimum, mean, and median values of g.
5. Create NumPy arrays to store the matrices

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 2 & 4 & 6 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 5 & 0 \\ 0 & 1 & 1 \\ 4 & 3 & 1 \end{pmatrix} \tag{2}$$

   Calculate $C = A + 2B$. Then compute AB and BA. You should notice that $AB \neq BA$.
6. A common operation to perform on matrices is to turn the rows into columns and the columns into rows. This is called transposing the matrix. Use the function `transpose` to compute the transpose of A and print the result.
7. Solve the linear system of equations $Ax = b$ where

$$A = \begin{pmatrix} 1 & 0 & 0 & -1 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 2 & 0 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 2 \end{pmatrix} \tag{3}$$

   Print the array x. Then compute $Ax - b$ and print the result.

Exercise 3 - Weather prediction (Essential)

In this example we'll use a **Markov chain** to create a simple model for weather prediction. To start, we will assume that there are three states of weather: sunny, cloudy, and rainy. We will use the state vector $x = (x_0, x_1, x_2)$ to describe the probabilities of the weather being sunny (x_0), cloudy (x_1), or rainy (x_2). We use a transition matrix P to describe how the weather changes from one day to the next. The entries of the transition matrix, $P_{i,j}$, describe the probability of going from state i to state j. For this problem, we will assume that

$$P = \begin{pmatrix} 0.5 & 0.3 & 0.2 \\ 0.4 & 0.2 & 0.4 \\ 0.6 & 0.2 & 0.2 \end{pmatrix} \tag{4}$$

The entry $P_{1,1} = 0.5$ means there is a 50% chance that if a day is sunny, then the next day will be sunny. Similarly, $P_{3,1} = 0.6$ means there is a 60% chance that if a day is rainy, then the next day will be sunny.

1. Suppose that today is sunny. Then we can write the state vector as $x^{(0)} = (1, 0, 0)$. The weather tomorrow can be predicted by computing the product $x^{(1)} = x^{(0)}P$. What is the probability that tomorrow will be sunny?
2. The product $x^{(2)} = x^{(1)}P = x^{(0)}P^2$ can be used to predict the weather in two days. What is the probability that it will rain in two days?
3. Provide a prediction of the weather for seven days from now. That is, compute $x^{(7)}$.