

Advanced Programming

Samir Datta*

January 29, 2018

Instructions

1. Modify the attached file `assignment.1.py` as *yourusername.1.py*. For example, I would have used `aalok.1.py`.
 2. The file that you submit should be a modification of the attached `assignment.1.py` file, i.e, it should contain the functions that have already been defined in this file. You may want to define more, but the given ones, including `main` must be there.
 3. The functions provided will be treated as explained in the problem statements and the comments. You are free to change the names for the arguments, but the code will be tested by running the `main` function in the file (after we add some more test cases).
 4. The *return* type of each function should be as specified in the `assignment.1.py` file.
-

Problem 1

Implement the function **matrixMultiply** that takes two matrices of appropriate dimensions and returns their product. Each list should be interpreted as a row of the matrix.

Note:

$$\begin{bmatrix} 1 & 2 & 5 & 4 \\ 4 & 6 & 7 & 8 \\ 9 & 0 & 1 & 7 \end{bmatrix}$$

will be encoded as `[[1, 2, 5, 4], [4, 6, 7, 8], [9, 0, 1, 7]]`

Assume:

- The matrix given as input will always be a valid matrix.
- The input matrices will be of appropriate dimensions and will be compatible with multiplication.

*Teaching Assistants - Kishlaya Jaiswal, Agnishom Chattopadhyay

Problem 2

Given k numbers $seed[0], seed[1], seed[2], \dots, seed[k-1]$, define

$$kNacci(k, seed, n) = \begin{cases} \sum_{i=1}^k kNacci(k, seed, n-i) & n \geq k \\ seed[n] & \text{otherwise.} \end{cases}$$

Implement the function **kNacci(k, seed, n)** that computes this.

Assume:

- k is positive and n is a non-negative integer.
- $seed$ is an array of size k of integers.

Problem 3

A group of $N \geq 1$ children labelled $[1..N]$ are playing a counting out game, with parameter $k \geq 1$ to select a leader among themselves. The game proceeds as follows:

- The children form a circle.
- Child 1 is marked as an invalid candidate.
- Ignoring the invalid children, the child who is k places to the right of child 1 is again marked invalid.
- The process is repeated until all but one child is marked invalid.
- The last remaining child is selected as the leader
- Here are the steps of the process run with 5 children and $k = 2$:

– 1 2 3 4 5
– 1 2 3 4 5
– 1 2 3 4 5
– 1 2 3 4 5
– 1 2 3 4 5

Implement **selectLeader(N, k)** that returns the run of the game. In the output, each sublist should be in ascending order.

Problem 4

Let $B \geq 3$ be a positive integer. Implement **representable(B, n)** which is *True* if $n \in A$ and *False* otherwise. A is defined as follows:

$$A_0 = \{1\}$$

$$A_i = \{B \cdot m, B \cdot m + 1, B \cdot m + 2, \dots, B \cdot m + (B - 2) \mid m \in A_{i-1}\} \quad \text{for } i > 0$$

$$A = \bigcup_{i \geq 0} A_i$$

welcome to the flying circus:

all my puns are indented.