X

 (https://swayam.gov.in)  (https://swayam.gov.in/nc_details/NPTEL)

adarshkumar8225@gmail.com

NPTEL (https://swayam.gov.in/explorer?ncCode=NPTEL) » Programming, Data Structures And Algorithms Using Python (course)

Announcements (announcements)

About the Course (https://swayam.gov.in/nd1_noc20_cs26/preview) Ask a Question (forum)

Progress (student/home) Mentor (student/mentor)

Course outline

- **How does an NPTEL online course work?**
- **Week 1 : Introduction**
- **Week 1 Quiz**
- **Week 2: Basics of Python**
- **Week 2 Quiz**
- **Week 2 Programming Assignment**
- **Week 3: Lists, inductive function**

Week 3 Programming Assignment

Due on 2020-02-20, 23:59 IST

Write three Python functions as specified below. Paste the text for all three functions together into the submission window.

- You may define additional auxiliary functions as needed.
- In all cases you may assume that the value passed to the function is of the expected type, so your function does not have to check for malformed inputs.
- For each function, there are some public test cases and some (hidden) private test cases.
- "Compile and run" will evaluate your submission against the public test cases.
- "Submit" will evaluate your submission against the hidden private test cases and report a score on 100. There are 10 private testcases in all, each with equal weightage.
- Ignore warnings about "Presentation errors".

1. Define a Python function `remdup(l)` that takes a nonempty list of integers `l` and removes all duplicates in `l`, keeping only the first occurrence of each number. For instance:

```
>>> remdup([3,1,3,5])
[3, 1, 5]

>>> remdup([7,3,-1,-5])
```

```
[7, 3, -1, -5]

>>> remdup([3,5,7,5,3,7,10])
[3, 5, 7, 10]
```

2. Write a Python function `sumsquare(l)` that takes a nonempty list of integers and returns a list `[odd,even]`, where `odd` is the sum of squares all the odd numbers in `l` and `even` is the sum of squares of all the even numbers in `l`.

   Here are some examples to show how your function should work.

```
>>> sumsquare([1,3,5])
[35, 0]

>>> sumsquare([2,4,6])
[0, 56]

>>> sumsquare([-1,-2,3,7])
[59, 4]
```

3. A two dimensional matrix can be represented in Python row-wise, as a list of lists: each inner list represents one row of the matrix. For instance, the matrix

```
1  2  3  4
5  6  7  8
```

   would be represented as `[[1, 2, 3, 4], [5, 6, 7, 8]]`.

   The transpose of a matrix converts each row into a column. The transpose of the matrix above is:

```
1  5
2  6
3  7
4  8
```

   which would be represented as `[[1, 5], [2, 6], [3, 7], [4, 8]]`.

   Write a Python function `transpose(m)` that takes as input a two dimensional matrix `m` and returns the transpose of `m`. **The argument `m` should remain undisturbed by the function.**

   Here are some examples to show how your function should work. You may assume that the input to the function is always a non-empty matrix.

```
>>> transpose([[1,2,3],[4,5,6]])
[[1, 4], [2, 5], [3, 6]]
```