X

(https://swayam-uat-central.appspot.com) (https://swayam-uat-central.appspot.com/nc_details/AICTE)

suchetajjw47@gmail.com

AICTE (https://swayam-uat-central.appspot.com/explorer?ncCode=AICTE) » Programming and Data Structures with Python (course)

Course outline

Practice Assignments

Practice Quiz 1

Quiz 1, Mon 25 Oct 2021

PDSP Assignment 1, due Tue 2 Nov 2021

- **PDSP Assignment 1 (/programming_2021/progassignment?name=11)**

PDSP Assignment 2, due Fri 12 Nov 2021

Quiz 2, Mon 8 Nov 2021

PDSP Assignment 3, due Wed 24 Nov 2021

PDSP Assignment 1

Due on 2021-11-02, 23:59 IST

Write two Python functions as specified below. Paste the text for both functions together into the submission window. Your function will be called automatically with various inputs and should return values as specified. Do not write commands to read any input or print any output.

- 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 normally 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. There are 10 private test cases, with equal weightage. You will get feedback about which private test cases pass or fail, though you cannot see the actual test cases.
- Ignore warnings about "Presentation errors".

1. Write a Python function `histogram(l)` that takes as input a list of integers with repetitions and returns a list of pairs as follows:.
    - for each number `n` that appears in `l`, there should be exactly one pair `(n,r)` in the list returned by the function, where `r` is the number of repetitions of `n` in `l`.
    - the final list should be sorted in ascending order by `r`, the number of repetitions. For numbers that occur with the same number of repetitions, arrange the pairs in ascending order of the value of the number.

    For instance:

PDSP Assignment 4, due Fri 17 Dec 2021

Quiz 3, Thu 16 Dec 2021

PDSP Quiz 4, Thu 23 Dec 2021

PDSP Assignment 5, due Fri 31 Dec 2021

For instance:

```
>>> histogram([13,12,11,13,14,13,7,7,13,14,12])
[(11, 1), (7, 2), (12, 2), (14, 2), (13, 4)]

>>> histogram([7,12,11,13,7,11,13,14,12])
[(14, 1), (7, 2), (11, 2), (12, 2), (13, 2)]

>>> histogram([13,7,12,7,11,13,14,13,7,11,13,14,12,14,14,7])
[(11, 2), (12, 2), (7, 4), (13, 4), (14, 4)]
```

2. A college maintains academic information about students in three separate lists
   - Course details: A list of pairs of form (coursecode,coursename), where both entries are strings. For instance, `[ ("MA101","Calculus"),("PH101","Mechanics"), ("HU101","English") ]`
   - Student details: A list of pairs of form (rollnumber,name), where both entries are strings. For instance, `[ ("UGM2018001","Rohit Grewal"),("UGP2018132","Neha Talwar") ]`
   - A list of triples of the form (rollnumber,coursecode,grade), where all entries are strings. For instance, `[ ("UGM2018001", "MA101", "AB"), ("UGP2018132", "PH101", "B"), ("UGM2018001", "PH101", "B") ]`. You may assume that each roll number and course code in the grade list appears in the student details and course details, respectively.

   Your task is to write a function transcript (coursedetails,studentdetails,grades) that takes these three lists as input and produces consolidated grades for each student. Each of the input lists may have its entries listed in arbitrary order. Each entry in the returned list should be a tuple of the form

   `(rollnumber, name,[(coursecode_1,coursename_1,grade_1),..., (coursecode_k,coursename_k,grade_k)])`

   where the student has grades for k >= 1 courses reported in the input list grades.

   The output list should be organized as follows.

   - The tuples shold sorted in ascending order by rollnumber
   - Each student's grades should sorted in ascending order by coursecode

   For instance

```
>>>transcript([("MA101","Calculus"),("PH101","Mechanics"),("HU10
1","English")],[("UGM2021001","Rohit Grewal"),("UGP2021132","Neha
Talwar")],[("UGM2021001","MA101","AB"),("UGP2021132","PH10
1","B"),("UGM2021001","PH101","B")])
```

```
1","B"),("UGM2021001","PH101","B")])

[('UGM2021001', 'Rohit Grewal', [('MA101', 'Calculus', 'AB'), ('PH101', 'Mechanics', 'B')]), ('UGP2021132', 'Neha Talwar', [('PH101', 'Mechanics', 'B')])]

>>>transcript([("T1","Test 1"),("T2","Test 2"),("T3","Test 3")],[("Opener","Rohit Sharma"),("Captain","Virat Kohli"),("No3","Cheteshwar Pujara")],[("Opener","T1","14"),("Captain","T1","33"),("No3","T1","30"),("Opener","T2","55") ,("Captain","T2","158"),("No3","T2","19"), ("Opener","T3","33"),("Captain","T3","95"),("No3","T3","51")])

[('Captain', 'Virat Kohli', [('T1', 'Test 1', '33'), ('T2', 'Test 2', '158'), ('T3', 'Test 3', '95')]), ('No3', 'Cheteshwar Pujara', [('T1', 'Test 1', '30'), ('T2', 'Test 2', '19'), ('T3', 'Test 3', '51')]), ('Opener', 'Rohit Sharma', [('T1', 'Test 1', '14'), ('T2', 'Test 2', '55'), ('T3', 'Test 3', '33')])]
```

Private Test cases used for evaluation	Input	Expected Output	Actual Output	Status
Test Case 1	`histogram([310 ,310 ,310 ,310 ,310 ,970 ,970 ,970 ,970 ,970 ,770 ,770 ,906 ,906 ,906 ,906 ,906 ,906 ,906 ,906 ,906 ,199 ,199 ,199 ,199 ,199 ,199 ,199 ,866 ,866 ,866 ,866 ,866 ,866 ,866 ,866 ,866 ,866 ,314 ,314 ,314 ,314 ,314 ,314 ,966 ,966 ,966 ,966 ,695 ,695 ,695 ,695 ,695 ,695 ,695 ,695 ,695 ,695 ,359 ,359 ,359 ,961 ,961 ,961 ,961 ,961 ,961 ,961 ,801 ,801 ,839 ,839 ,839 ,839 ,574 ,574 ,322 ,322 ,322 ,322 ,322 ,322 ,322 ,322 ,322 ,322 ,506 ,506 ,506 ,602 ,602 ,602 ,614 ,614 ,614 ,614 ,614 ,614 ,614 ,614 ,870 ,870 ,870 ,870 ,870 ,870`	`[(135, 2), (320, 2), (533, 2), (574, 2), (655, 2), (770, 2), (801, 2), (901, 2), (930, 2), (968, 2), (73, 3), (76, 3), (316, 3), (359, 3), (477, 3), (506, 3), (536, 3), (554, 3), (602, 3), (675, 3), (865, 3), (144, 4), (381, 4), (483, 4), (498, 4), (592, 4), (966, 4), (123, 5), (141, 5), (196, 5), (197, 5), (198, 5), (310, 5), (313, 5), (436, 5), (970, 5), (58, 6), (90, 6), (314, 6), (486, 6), (579, 6), (830, 6),`	`[(135, 2), (320, 2), (533, 2), (574, 2), (655, 2), (770, 2), (801, 2), (901, 2), (930, 2), (968, 2), (73, 3), (76, 3), (316, 3), (359, 3), (477, 3), (506, 3), (536, 3), (554, 3), (602, 3), (675, 3), (865, 3), (144, 4), (381, 4), (483, 4), (498, 4), (592, 4), (966, 4), (123, 5), (141, 5), (196, 5), (197, 5), (198, 5), (310, 5), (313, 5), (436, 5), (970, 5), (58, 6), (90, 6), (314, 6), (486, 6), (579, 6), (830, 6),`	Passed

,870 ,870 ,381 ,381 ,381 ,381 ,533 ,533 ,675 ,675 ,675 ,480 ,480 ,480 ,480 ,480 ,480 ,480 ,772 ,772 ,772 ,772 ,772 ,772 ,772 ,772 ,772 ,218 ,218 ,218 ,218 ,218 ,218 ,218 ,218 ,197 ,197 ,197 ,197 ,197 ,141 ,141 ,141 ,141 ,141 ,123 ,123 ,123 ,123 ,123 ,689 ,689 ,689 ,689 ,689 ,689 ,689 ,689 ,689 ,606 ,606 ,606 ,606 ,606 ,606 ,606 ,41 ,41 ,41 ,41 ,41 ,41 ,41 ,41 ,41 ,839 ,839 ,811 ,811 ,811 ,811 ,811 ,811 ,705 ,705 ,705 ,705 ,705 ,705 ,705 ,705 ,705 ,705 ,584 ,584 ,584 ,584 ,584 ,584 ,584 ,584 ,579 ,579 ,579 ,579 ,579 ,579 ,90 ,90 ,90 ,90 ,90 ,90 ,99 ,99 ,99 ,99 ,99 ,99 ,99 ,99 ,491 ,491 ,491 ,491 ,491 ,491 ,491 ,642 ,642 ,642 ,642 ,642 ,642 ,642 ,486 ,486 ,486 ,486 ,486 ,486 ,852 ,852 ,852 ,852 ,852 ,852 ,852 ,852 ,300 ,300 ,300 ,300 ,300 ,300 ,300 ,619 ,619 ,619 ,619 ,619 ,619 ,619 ,554 ,554 ,554 ,429 ,429 ,429 ,429 ,689 ,689 ,689 ,689 ,689 ,689 ,689 ,320 ,320 ,592 ,592 ,592 ,592 ,196 ,196 ,196 ,196 ,196 ,811 ,811 ,811 ,811 ,811 ,436 ,436 ,436 ,436 ,436 ,520 ,520 ,520 ,520 ,520 ,520 ,520 ,865 ,865 ,865 ,147 ,147 ,147 ,147 ,147 ,147 ,147	(839, 6), (847, 6), (199, 7), (300, 7), (480, 7), (491, 7), (520, 7), (606, 7), (619, 7), (642, 7), (668, 7), (961, 7), (99, 8), (218, 8), (219, 8), (490, 8), (584, 8), (614, 8), (765, 8), (833, 8), (852, 8), (870, 8), (41, 9), (406, 9), (416, 9), (468, 9), (701, 9), (772, 9), (906, 9), (8, 10), (36, 10), (147, 10), (277, 10), (322, 10), (429, 10), (695, 10), (705, 10), (866, 10), (811, 11), (689, 16)]\n	(839, 6), (847, 6), (199, 7), (300, 7), (480, 7), (491, 7), (520, 7), (606, 7), (619, 7), (642, 7), (668, 7), (961, 7), (99, 8), (218, 8), (219, 8), (490, 8), (584, 8), (614, 8), (765, 8), (833, 8), (852, 8), (870, 8), (41, 9), (406, 9), (416, 9), (468, 9), (701, 9), (772, 9), (906, 9), (8, 10), (36, 10), (147, 10), (277, 10), (322, 10), (429, 10), (695, 10), (705, 10), (866, 10), (811, 11), (689, 16)]\n

	,147 ,147 ,147 ,73 ,73 ,73 ,76 ,76 ,76 ,406 ,406 ,406 ,406 ,406 ,406 ,406 ,406 ,406 ,313 ,313 ,313 ,313 ,313 ,847 ,847 ,847 ,847 ,847 ,847 ,198 ,198 ,198 ,198 ,198 ,930 ,930 ,536 ,536 ,536 ,316 ,316 ,316 ,416 ,416 ,416 ,416 ,416 ,416 ,416 ,416 ,416 ,765 ,765 ,765 ,765 ,765 ,765 ,765 ,765 ,468 ,468 ,468 ,468 ,468 ,468 ,468 ,468 ,468 ,833 ,833 ,833 ,833 ,833 ,833 ,833 ,833 ,58 ,58 ,58 ,58 ,58 ,58 ,36 ,36 ,36 ,36 ,36 ,36 ,36 ,36 ,36 ,36 ,498 ,498 ,498 ,498 ,144 ,144 ,144 ,144 ,483 ,483 ,483 ,483 ,830 ,830 ,830 ,830 ,830 ,830 ,701 ,701 ,701 ,701 ,701 ,701 ,701 ,701 ,701 ,655 ,655 ,490 ,490 ,490 ,490 ,490 ,490 ,490 ,490 ,135 ,135 ,429 ,429 ,429 ,429 ,429 ,429 ,8 ,8 ,8 ,8 ,8 ,8 ,8 ,8 ,8 ,8 ,901 ,901 ,219 ,219 ,219 ,219 ,219 ,219 ,219 ,219 ,668 ,668 ,668 ,668 ,668 ,668 ,668 ,968 ,968 ,477 ,477 ,477 ,277 ,277 ,277 ,277 ,277 ,277 ,277 ,277 ,277 ,277 ,])			
Test Case 2	histogram([123 ,123 ,123 ,123 ,123 ,9 ,9 ,397 ,397 ,397 ,397 ,397 ,397 ,397 ,397 ,397 ,32 ,32 ,32 ,32 ,32 ,32 ,447 ,447 ,447 ,447 ,447 ,447 ,447 ,583 ,583 ,583 ,583 ,583 ,583 ,829	[(9, 2), (213, 2), (352, 2), (574, 2), (739, 2), (829, 2), (905, 2), (157, 3), (206, 3), (262, 3), (360, 3), (378, 3), (802, 3), (117,	[(9, 2), (213, 2), (352, 2), (574, 2), (739, 2), (829, 2), (905, 2), (157, 3), (206, 3), (262, 3), (360, 3), (378, 3), (802, 3), (117, 4),	Passed

```
,829 ,407 ,407 ,407
,407 ,407 ,407 ,407
,407 ,53 ,53 ,53 ,53
,53 ,962 ,962 ,962
,962 ,962 ,962 ,962
,694 ,694 ,694 ,694
,694 ,694 ,694 ,397
,397 ,397 ,397 ,397
,397 ,397 ,397 ,397
,397 ,168 ,168 ,168
,354 ,354 ,354 ,354
,354 ,354 ,354 ,354
,354 ,623 ,623 ,623
,623 ,623 ,623 ,623
,623 ,221 ,221 ,221
,221 ,440 ,440 ,440
,440 ,440 ,739 ,739
,378 ,378 ,378 ,458
,458 ,458 ,458 ,458
,458 ,412 ,412 ,412
,412 ,412 ,412 ,412
,412 ,814 ,814 ,814
,297 ,297 ,297 ,297
,297 ,297 ,297 ,297
,297 ,295 ,295 ,295
,295 ,295 ,295 ,701
,701 ,701 ,701 ,701
,701 ,397 ,397 ,397
,397 ,397 ,397 ,397
,397 ,786 ,786 ,786
,786 ,108 ,108 ,108
,108 ,108 ,108 ,262
,262 ,262 ,33 ,33
,33 ,33 ,33 ,33 ,905
,905 ,393 ,393 ,393
,393 ,393 ,206 ,206
,206 ,407 ,407 ,407
,407 ,407 ,407 ,407
,407 ,407 ,407 ,861
,861 ,861 ,861 ,861
,861 ,861 ,564 ,564
,564 ,564 ,564 ,564
,10 ,10 ,10 ,10 ,10
,10 ,10 ,10 ,10 ,10
,634 ,634 ,634 ,634
,634 ,663 ,663 ,663
,663 ,663 ,663 ,663
,663 ,492 ,492 ,492
,492 ,802 ,802 ,802
,850 ,850 ,850 ,850
,850 ,850 ,850 ,850
,850 ,850 ,112 ,112
,112 ,112 ,112 ,112
```

```
4), (221, 4),
(492, 4), (786,
4), (53, 5), (86,
5), (123, 5),
(393, 5), (440,
5), (634, 5),
(747, 5), (31, 6),
(33, 6), (108, 6),
(295, 6), (329,
6), (351, 6),
(446, 6), (458,
6), (508, 6),
(525, 6), (549,
6), (564, 6),
(583, 6), (630,
6), (641, 6),
(701, 6), (726,
6), (903, 6),
(985, 6), (34, 7),
(59, 7), (90, 7),
(137, 7), (447,
7), (694, 7),
(735, 7), (861,
7), (918, 7),
(962, 7), (412,
8), (468, 8),
(590, 8), (663,
8), (831, 8),
(943, 8), (112,
9), (168, 9),
(297, 9), (354,
9), (535, 9),
(538, 9), (555,
9), (670, 9),
(709, 9), (811,
9), (910, 9), (10,
10), (331, 10),
(728, 10), (850,
10), (887, 10),
(895, 10), (814,
12), (298, 13),
(32, 16), (623,
17), (407, 18),
(397, 27)]\n
```

```
(221, 4), (492,
4), (786, 4), (53,
5), (86, 5), (123,
5), (393, 5),
(440, 5), (634,
5), (747, 5), (31,
6), (33, 6), (108,
6), (295, 6),
(329, 6), (351,
6), (446, 6),
(458, 6), (508,
6), (525, 6),
(549, 6), (564,
6), (583, 6),
(630, 6), (641,
6), (701, 6),
(726, 6), (903,
6), (985, 6), (34,
7), (59, 7), (90,
7), (137, 7),
(447, 7), (694,
7), (735, 7),
(861, 7), (918,
7), (962, 7),
(412, 8), (468,
8), (590, 8),
(663, 8), (831,
8), (943, 8),
(112, 9), (168,
9), (297, 9),
(354, 9), (535,
9), (538, 9),
(555, 9), (670,
9), (709, 9),
(811, 9), (910,
9), (10, 10),
(331, 10), (728,
10), (850, 10),
(887, 10), (895,
10), (814, 12),
(298, 13), (32,
16), (623, 17),
(407, 18), (397,
27)]\n
```

```
,112 ,112 ,112 ,887
,887 ,887 ,887 ,887
,887 ,887 ,887 ,887
,887 ,814 ,814 ,814
,814 ,814 ,814 ,814
,814 ,814 ,910 ,910
,910 ,910 ,910 ,910
,910 ,910 ,910 ,213
,213 ,90 ,90 ,90 ,90
,90 ,90 ,90 ,32 ,32
,32 ,32 ,32 ,32 ,32
,32 ,32 ,32 ,525
,525 ,525 ,525 ,525
,525 ,726 ,726 ,726
,726 ,726 ,726 ,168
,168 ,168 ,168 ,168
,168 ,831 ,831 ,831
,831 ,831 ,831 ,831
,831 ,549 ,549 ,549
,549 ,549 ,549 ,298
,298 ,298 ,298 ,298
,298 ,298 ,811 ,811
,811 ,811 ,811 ,811
,811 ,811 ,811 ,508
,508 ,508 ,508 ,508
,508 ,555 ,555 ,555
,555 ,555 ,555 ,555
,555 ,555 ,641 ,641
,903 ,903 ,903 ,903
,903 ,903 ,86 ,86
,86 ,86 ,86 ,590
,590 ,590 ,590 ,590
,590 ,590 ,590 ,538
,538 ,538 ,538 ,538
,538 ,538 ,538 ,538
,331 ,331 ,331 ,331
,331 ,331 ,331 ,331
,331 ,331 ,943 ,943
,943 ,943 ,943 ,943
,943 ,943 ,670 ,670
,670 ,670 ,670 ,670
,670 ,670 ,670 ,34
,34 ,34 ,34 ,34 ,34
,34 ,735 ,735 ,735
,735 ,735 ,735 ,735
,630 ,630 ,630 ,630
,630 ,630 ,352 ,352
,298 ,298 ,298 ,298
,298 ,298 ,59 ,59
,59 ,59 ,59 ,59 ,59
,117 ,117 ,117 ,117
,895 ,895 ,895 ,895
,895 ,895 ,895 ,895
```

	,895 ,895 ,446 ,446 ,446 ,446 ,446 ,446 ,623 ,623 ,623 ,623 ,623 ,623 ,623 ,623 ,623 ,709 ,709 ,709 ,709 ,709 ,709 ,709 ,709 ,709 ,31 ,31 ,31 ,31 ,31 ,31 ,728 ,728 ,728 ,728 ,728 ,728 ,728 ,728 ,728 ,728 ,137 ,137 ,137 ,137 ,137 ,137 ,137 ,641 ,641 ,641 ,641 ,329 ,329 ,329 ,329 ,329 ,329 ,574 ,574 ,535 ,535 ,535 ,535 ,535 ,535 ,535 ,535 ,535 ,918 ,918 ,918 ,918 ,918 ,918 ,918 ,351 ,351 ,351 ,351 ,351 ,351 ,157 ,157 ,157 ,747 ,747 ,747 ,747 ,747 ,360 ,360 ,360 ,468 ,468 ,468 ,468 ,468 ,468 ,468 ,468 ,985 ,985 ,985 ,985 ,985 ,985 ,])			
Test Case 3	histogram([240 ,240 ,240 ,240 ,240 ,240 ,240 ,240 ,354 ,354 ,291 ,291 ,291 ,291 ,291 ,291 ,840 ,840 ,840 ,840 ,840 ,840 ,664 ,664 ,664 ,597 ,597 ,597 ,597 ,597 ,597 ,597 ,622 ,622 ,622 ,580 ,580 ,910 ,910 ,910 ,910 ,910 ,910 ,910 ,910 ,910 ,247 ,247 ,247 ,247 ,247 ,247 ,247 ,247 ,247 ,28 ,28 ,909 ,909 ,909 ,909 ,909 ,934 ,934 ,934 ,899 ,899 ,899 ,899 ,899 ,899 ,569 ,569 ,569 ,569 ,569 ,859 ,859 ,859 ,154 ,154 ,154 ,154 ,154 ,154 ,154 ,154 ,154 ,239 ,239 ,239 ,239 ,239 ,239 ,239 ,239 ,239 ,239	[(28, 2), (204, 2), (317, 2), (354, 2), (466, 2), (580, 2), (889, 2), (969, 2), (176, 3), (308, 3), (341, 3), (404, 3), (458, 3), (513, 3), (515, 3), (622, 3), (664, 3), (859, 3), (930, 3), (934, 3), (19, 4), (212, 4), (234, 4), (450, 4), (501, 4), (61, 5), (92, 5), (366, 5), (569, 5), (866, 5), (909, 5), (72, 6), (291, 6), (524, 6), (840, 6), (899, 6), (232, 7), (431, 7), (597, 7),	[(28, 2), (204, 2), (317, 2), (354, 2), (466, 2), (580, 2), (889, 2), (969, 2), (176, 3), (308, 3), (341, 3), (404, 3), (458, 3), (513, 3), (515, 3), (622, 3), (664, 3), (859, 3), (930, 3), (934, 3), (19, 4), (212, 4), (234, 4), (450, 4), (501, 4), (61, 5), (92, 5), (366, 5), (569, 5), (866, 5), (909, 5), (72, 6), (291, 6), (524, 6), (840, 6), (899, 6), (232, 7), (431, 7), (597, 7),	Passed

,969 ,969 ,366 ,366 ,366 ,366 ,366 ,602 ,602 ,602 ,602 ,602 ,602 ,602 ,453 ,453 ,453 ,453 ,453 ,453 ,453 ,453 ,884 ,884 ,884 ,884 ,884 ,884 ,884 ,884 ,456 ,456 ,456 ,456 ,456 ,456 ,456 ,866 ,866 ,866 ,977 ,977 ,212 ,212 ,212 ,212 ,431 ,431 ,431 ,431 ,431 ,431 ,431 ,72 ,72 ,72 ,72 ,72 ,72 ,247 ,247 ,247 ,247 ,866 ,866 ,48 ,48 ,48 ,48 ,48 ,48 ,48 ,48 ,48 ,48 ,232 ,232 ,232 ,232 ,232 ,232 ,232 ,404 ,404 ,404 ,317 ,317 ,349 ,349 ,349 ,349 ,349 ,349 ,349 ,349 ,349 ,456 ,456 ,456 ,456 ,456 ,456 ,456 ,456 ,456 ,216 ,216 ,216 ,216 ,216 ,216 ,216 ,216 ,513 ,513 ,513 ,662 ,662 ,662 ,662 ,662 ,662 ,662 ,662 ,567 ,567 ,567 ,567 ,567 ,567 ,567 ,567 ,567 ,567 ,501 ,501 ,501 ,501 ,605 ,605 ,605 ,605 ,605 ,605 ,605 ,605 ,605 ,920 ,920 ,920 ,920 ,920 ,920 ,920 ,458 ,458 ,458 ,176 ,176 ,176 ,930 ,930 ,930 ,411 ,411 ,411 ,411 ,411 ,411 ,411 ,411 ,411 ,411 ,61 ,61 ,61 ,61 ,61 ,308 ,308 ,308 ,450 ,450 ,450 ,450 ,341 ,341 ,341 ,338 ,338 ,338 ,338 ,338 ,338 ,338 ,338 ,338 ,338 ,306 ,306 ,306 ,306 ,306 ,306 ,306 ,306 ,306 ,306 ,524 ,524 ,524 ,524 ,524 ,524 ,595	(602, 7), (920, 7), (979, 7), (216, 8), (240, 8), (453, 8), (662, 8), (884, 8), (49, 9), (154, 9), (349, 9), (495, 9), (595, 9), (605, 9), (637, 9), (910, 9), (13, 10), (48, 10), (133, 10), (209, 10), (239, 10), (258, 10), (306, 10), (338, 10), (411, 10), (567, 10), (834, 10), (933, 10), (977, 12), (247, 13), (456, 16)]\n	(602, 7), (920, 7), (979, 7), (216, 8), (240, 8), (453, 8), (662, 8), (884, 8), (49, 9), (154, 9), (349, 9), (495, 9), (595, 9), (605, 9), (637, 9), (910, 9), (13, 10), (48, 10), (133, 10), (209, 10), (239, 10), (258, 10), (306, 10), (338, 10), (411, 10), (567, 10), (834, 10), (933, 10), (977, 12), (247, 13), (456, 16)]\n

	,595 ,595 ,595 ,595 ,595 ,595 ,595 ,595 ,133 ,133 ,133 ,133 ,133 ,133 ,133 ,133 ,133 ,133 ,977 ,977 ,977 ,977 ,977 ,977 ,977 ,977 ,977 ,977 ,19 ,19 ,19 ,19 ,466 ,466 ,92 ,92 ,92 ,92 ,92 ,13 ,13 ,13 ,13 ,13 ,13 ,13 ,13 ,13 ,13 ,933 ,933 ,933 ,933 ,933 ,933 ,933 ,933 ,933 ,933 ,889 ,889 ,234 ,234 ,234 ,234 ,209 ,209 ,209 ,209 ,209 ,209 ,209 ,209 ,209 ,209 ,258 ,258 ,258 ,258 ,258 ,258 ,258 ,258 ,258 ,258 ,204 ,204 ,637 ,637 ,637 ,637 ,637 ,637 ,637 ,637 ,637 ,495 ,495 ,495 ,495 ,495 ,495 ,495 ,495 ,495 ,834 ,834 ,834 ,834 ,834 ,834 ,834 ,834 ,834 ,834 ,515 ,515 ,515 ,49 ,49 ,49 ,49 ,49 ,49 ,49 ,49 ,49 ,979 ,979 ,979 ,979 ,979 ,979 ,979 ,])			
Test Case 4	histogram([936 ,936 ,936 ,936 ,936 ,936 ,936 ,936 ,131 ,131 ,131 ,131 ,83 ,83 ,83 ,83 ,83 ,83 ,83 ,486 ,486 ,486 ,486 ,486 ,486 ,75 ,75 ,75 ,75 ,75 ,594 ,594 ,594 ,594 ,594 ,594 ,594 ,215 ,215 ,215 ,215 ,215 ,215 ,215 ,215 ,215 ,845 ,845 ,845 ,845 ,845 ,845 ,845 ,845 ,189 ,189 ,189 ,189 ,189 ,189 ,189 ,189 ,189 ,861 ,861 ,861 ,861 ,861 ,861 ,861 ,350 ,350 ,645 ,645 ,645	[(37, 2), (269, 2), (350, 2), (516, 2), (660, 2), (911, 2), (935, 2), (994, 2), (197, 3), (239, 3), (252, 3), (368, 3), (487, 3), (514, 3), (537, 3), (565, 3), (617, 3), (622, 3), (645, 3), (30, 4), (131, 4), (238, 4), (302, 4), (385, 4), (548, 4), (75, 5), (198, 5), (210, 5), (265, 5), (292,	[(37, 2), (269, 2), (350, 2), (516, 2), (660, 2), (911, 2), (935, 2), (994, 2), (197, 3), (239, 3), (252, 3), (368, 3), (487, 3), (514, 3), (537, 3), (565, 3), (617, 3), (622, 3), (645, 3), (30, 4), (131, 4), (238, 4), (302, 4), (385, 4), (548, 4), (75, 5), (198, 5), (210, 5), (265, 5), (292,	Passed

,617 ,617 ,617 ,302 ,302 ,302 ,302 ,444 ,444 ,444 ,444 ,444 ,444 ,444 ,444 ,444 ,444 ,406 ,406 ,406 ,406 ,406 ,406 ,406 ,406 ,163 ,163 ,163 ,163 ,163 ,163 ,163 ,793 ,793 ,793 ,793 ,793 ,793 ,793 ,793 ,257 ,257 ,257 ,257 ,257 ,257 ,257 ,385 ,385 ,385 ,385 ,680 ,680 ,680 ,680 ,680 ,680 ,680 ,680 ,286 ,286 ,286 ,286 ,286 ,286 ,487 ,487 ,487 ,416 ,416 ,416 ,416 ,416 ,416 ,416 ,532 ,532 ,532 ,532 ,532 ,660 ,660 ,639 ,639 ,639 ,639 ,639 ,639 ,639 ,639 ,582 ,582 ,582 ,582 ,582 ,582 ,582 ,582 ,582 ,582 ,238 ,238 ,238 ,238 ,265 ,265 ,265 ,265 ,265 ,622 ,622 ,622 ,999 ,999 ,999 ,999 ,999 ,999 ,999 ,999 ,999 ,532 ,532 ,532 ,532 ,532 ,532 ,532 ,532 ,936 ,936 ,936 ,936 ,936 ,936 ,373 ,373 ,373 ,373 ,373 ,373 ,373 ,363 ,363 ,363 ,363 ,363 ,363 ,363 ,292 ,292 ,292 ,292 ,292 ,516 ,516 ,110 ,110 ,110 ,110 ,110 ,110 ,110 ,110 ,761 ,761 ,761 ,761 ,761 ,598 ,598 ,598 ,598 ,598 ,537 ,537 ,537 ,994 ,994 ,935 ,935 ,30 ,30 ,30 ,30 ,139 ,139 ,139 ,139 ,139 ,139 ,139 ,139 ,269 ,269 ,531 ,531 ,786 ,786 ,786 ,786 ,786 ,786 ,786 ,786 ,786 ,326 ,326 ,326 ,326 ,326 ,326 ,326 ,326	5), (598, 5), (642, 5), (661, 5), (761, 5), (865, 5), (286, 6), (476, 6), (486, 6), (722, 6), (784, 6), (83, 7), (163, 7), (257, 7), (363, 7), (373, 7), (416, 7), (459, 7), (594, 7), (715, 7), (861, 7), (110, 8), (138, 8), (139, 8), (324, 8), (326, 8), (367, 8), (382, 8), (406, 8), (639, 8), (680, 8), (732, 8), (793, 8), (845, 8), (930, 8), (126, 9), (189, 9), (215, 9), (390, 9), (519, 9), (762, 9), (999, 9), (217, 10), (264, 10), (444, 10), (457, 10), (582, 10), (531, 12), (786, 12), (532, 13), (936, 14)]\n	5), (598, 5), (642, 5), (661, 5), (761, 5), (865, 5), (286, 6), (476, 6), (486, 6), (722, 6), (784, 6), (83, 7), (163, 7), (257, 7), (363, 7), (373, 7), (416, 7), (459, 7), (594, 7), (715, 7), (861, 7), (110, 8), (138, 8), (139, 8), (324, 8), (326, 8), (367, 8), (382, 8), (406, 8), (639, 8), (680, 8), (732, 8), (793, 8), (845, 8), (930, 8), (126, 9), (189, 9), (215, 9), (390, 9), (519, 9), (762, 9), (999, 9), (217, 10), (264, 10), (444, 10), (457, 10), (582, 10), (531, 12), (786, 12), (532, 13), (936, 14)]\n

```
,326 ,476 ,476 ,476
,476 ,476 ,476 ,367
,367 ,367 ,367 ,367
,367 ,367 ,367 ,786
,786 ,786 ,37 ,37
,565 ,565 ,565 ,324
,324 ,324 ,324 ,324
,324 ,324 ,324 ,382
,382 ,382 ,382 ,382
,382 ,382 ,382 ,642
,642 ,642 ,642 ,642
,531 ,531 ,531 ,531
,531 ,531 ,531 ,531
,531 ,531 ,459 ,459
,459 ,459 ,459 ,459
,459 ,514 ,514 ,514
,911 ,911 ,930 ,930
,930 ,930 ,930 ,930
,930 ,930 ,264 ,264
,264 ,264 ,264 ,264
,264 ,264 ,264 ,264
,239 ,239 ,239 ,217
,217 ,217 ,217 ,217
,217 ,217 ,217 ,217
,217 ,784 ,784 ,784
,784 ,784 ,784 ,126
,126 ,126 ,126 ,126
,126 ,126 ,126 ,126
,865 ,865 ,865 ,865
,865 ,252 ,252 ,252
,390 ,390 ,390 ,390
,390 ,390 ,390 ,390
,390 ,138 ,138 ,138
,138 ,138 ,138 ,138
,138 ,732 ,732 ,732
,732 ,732 ,732 ,732
,732 ,210 ,210 ,210
,210 ,210 ,197 ,197
,197 ,457 ,457 ,457
,457 ,457 ,457 ,457
,457 ,457 ,457 ,198
,198 ,198 ,198 ,198
,715 ,715 ,715 ,715
,715 ,715 ,715 ,762
,762 ,762 ,762 ,762
,762 ,762 ,762 ,762
,548 ,548 ,548 ,548
,368 ,368 ,368 ,722
,722 ,722 ,722 ,722
,722 ,661 ,661 ,661
,661 ,661 ,519 ,519
,519 ,519 ,519 ,519
,519 ,519 ,519 ,])
```

Test Case 5	histogram([146 ,146 ,146 ,244 ,244 ,244 ,244 ,244 ,56 ,56 ,56 ,56 ,56 ,854 ,854 ,854 ,854 ,854 ,854 ,854 ,271 ,271 ,271 ,271 ,307 ,307 ,307 ,307 ,307 ,307 ,307 ,307 ,307 ,677 ,677 ,461 ,461 ,461 ,461 ,461 ,461 ,461 ,461 ,461 ,923 ,923 ,923 ,923 ,923 ,923 ,923 ,923 ,923 ,923 ,672 ,672 ,672 ,672 ,672 ,672 ,672 ,320 ,320 ,320 ,320 ,320 ,320 ,869 ,869 ,869 ,79 ,79 ,79 ,79 ,79 ,257 ,257 ,257 ,257 ,257 ,257 ,257 ,257 ,257 ,448 ,448 ,448 ,448 ,448 ,448 ,448 ,153 ,153 ,153 ,153 ,153 ,153 ,153 ,153 ,153 ,510 ,510 ,55 ,55 ,55 ,365 ,365 ,365 ,365 ,365 ,365 ,365 ,365 ,365 ,365 ,908 ,908 ,908 ,908 ,908 ,908 ,908 ,427 ,427 ,427 ,427 ,427 ,337 ,337 ,80 ,80 ,80 ,80 ,80 ,80 ,80 ,80 ,80 ,80 ,915 ,915 ,915 ,915 ,915 ,593 ,593 ,593 ,593 ,593 ,593 ,593 ,508 ,508 ,165 ,165 ,165 ,165 ,165 ,165 ,165 ,439 ,439 ,194 ,194 ,194 ,194 ,194 ,194 ,194 ,194 ,128 ,128 ,128 ,128 ,128 ,128 ,128 ,128 ,128 ,128 ,221 ,221 ,221 ,221 ,865 ,865 ,865 ,865 ,777 ,777 ,777 ,777 ,339 ,339 ,339 ,201 ,201 ,201 ,201 ,201 ,201 ,839 ,839 ,839 ,934 ,934 ,934 ,934	[(211, 2), (219, 2), (266, 2), (298, 2), (337, 2), (439, 2), (508, 2), (510, 2), (598, 2), (670, 2), (677, 2), (55, 3), (115, 3), (146, 3), (326, 3), (339, 3), (541, 3), (619, 3), (628, 3), (839, 3), (849, 3), (869, 3), (136, 4), (137, 4), (191, 4), (221, 4), (576, 4), (662, 4), (777, 4), (865, 4), (56, 5), (73, 5), (79, 5), (187, 5), (215, 5), (244, 5), (333, 5), (427, 5), (810, 5), (915, 5), (81, 6), (201, 6), (320, 6), (377, 6), (556, 6), (787, 6), (934, 6), (11, 7), (165, 7), (169, 7), (448, 7), (672, 7), (674, 7), (772, 7), (854, 7), (908, 7), (3, 8), (142, 8), (194, 8), (214, 8), (570, 8), (785, 8), (19, 9), (114, 9), (153, 9), (257, 9), (307, 9), (395, 9), (461, 9), (629, 9), (703, 9), (949, 9), (80, 10), (128, 10), (365, 10), (857, 10), (923, 10), (271, 11), (231, 14), (593, 16)]\n	[(211, 2), (219, 2), (266, 2), (298, 2), (337, 2), (439, 2), (508, 2), (510, 2), (598, 2), (670, 2), (677, 2), (55, 3), (115, 3), (146, 3), (326, 3), (339, 3), (541, 3), (619, 3), (628, 3), (839, 3), (849, 3), (869, 3), (136, 4), (137, 4), (191, 4), (221, 4), (576, 4), (662, 4), (777, 4), (865, 4), (56, 5), (73, 5), (79, 5), (187, 5), (215, 5), (244, 5), (333, 5), (427, 5), (810, 5), (915, 5), (81, 6), (201, 6), (320, 6), (377, 6), (556, 6), (787, 6), (934, 6), (11, 7), (165, 7), (169, 7), (448, 7), (672, 7), (674, 7), (772, 7), (854, 7), (908, 7), (3, 8), (142, 8), (194, 8), (214, 8), (570, 8), (785, 8), (19, 9), (114, 9), (153, 9), (257, 9), (307, 9), (395, 9), (461, 9), (629, 9), (703, 9), (949, 9), (80, 10), (128, 10), (365, 10), (857, 10), (923, 10), (271, 11), (231, 14), (593, 16)]\n	Passed

```
,934 ,934 ,114 ,114
,114 ,114 ,114 ,114
,114 ,114 ,114 ,187
,187 ,187 ,187 ,187
,598 ,598 ,619 ,619
,619 ,271 ,271 ,271
,271 ,271 ,271 ,271
,3 ,3 ,3 ,3 ,3 ,3 ,3
,3 ,211 ,211 ,219
,219 ,377 ,377 ,377
,377 ,377 ,377 ,674
,674 ,674 ,674 ,674
,674 ,674 ,19 ,19
,19 ,19 ,19 ,19 ,19
,19 ,19 ,576 ,576
,576 ,576 ,73 ,73
,73 ,73 ,73 ,810
,810 ,810 ,810 ,810
,395 ,395 ,395 ,395
,395 ,395 ,395 ,395
,395 ,541 ,541 ,541
,266 ,266 ,570 ,570
,570 ,570 ,570 ,570
,570 ,570 ,142 ,142
,142 ,142 ,142 ,142
,142 ,142 ,949 ,949
,949 ,949 ,949 ,949
,949 ,949 ,949 ,231
,231 ,231 ,231 ,231
,231 ,231 ,231 ,231
,231 ,785 ,785 ,785
,785 ,785 ,785 ,785
,785 ,11 ,11 ,11
,556 ,556 ,556 ,556
,556 ,556 ,703 ,703
,703 ,703 ,703 ,703
,703 ,703 ,703 ,191
,191 ,191 ,191 ,628
,628 ,628 ,169 ,169
,169 ,169 ,169 ,169
,169 ,231 ,231 ,231
,231 ,115 ,115 ,115
,137 ,137 ,137 ,137
,136 ,136 ,136 ,136
,787 ,787 ,787 ,787
,787 ,787 ,11 ,11
,11 ,11 ,857 ,857
,857 ,857 ,857 ,857
,857 ,857 ,857 ,857
,81 ,81 ,81 ,81 ,81
,81 ,593 ,593 ,593
,593 ,593 ,593 ,593
,593 ,593 ,214 ,214
```

	,214 ,214 ,214 ,214 ,214 ,214 ,849 ,849 ,849 ,662 ,662 ,662 ,662 ,333 ,333 ,333 ,333 ,333 ,629 ,629 ,629 ,629 ,629 ,629 ,629 ,629 ,629 ,670 ,670 ,772 ,772 ,772 ,772 ,772 ,772 ,772 ,298 ,298 ,215 ,215 ,215 ,215 ,215 ,326 ,326 ,326 ,])			
Test Case 6	transcript([('HV3', 'MP7SXFUU'), ('8J7', 'WQQYDXLU'), ('L2D', 'BR9EZ019'), ('YKQ', 'W8QAL4VU'), ('L74', '05NTPDLD'), ('QRH', '3NAQXDK5'), ('KGC', '8PFBI1NC'), ('2C0', 'COGOJ1U0')], [('2RZ9YC', 'HLNOAKSSJD95T45'), ('T1GNFU', 'X82J3MVZDA7YVL8'), ('C0BF8X', 'CR4S21F6YAMG3RQ'), ('SL2EWC', 'NQCY3NXRN5KM4GW'), ('GX8M2G', 'NUA13O7B1WRWPAM'), ('3HUBU4', '4SNO1JO1OAEB2ZZ'), ('9JVGIY', 'ETEAXDPZIXKYO97'), ('Y89DXJ', '0TKLC1UUDETFIL0'), ('O0VUFB', '2QD2CF2PVQWSOW6'), ('34Q54J', 'QT2TI1266JT1GLS')], [('GX8M2G', 'HV3', 'S'), ('3HUBU4', 'HV3', 'B'), ('Y89DXJ', 'HV3', 'F'), ('SL2EWC', '8J7', 'F'), ('GX8M2G', '8J7', 'B'), ('3HUBU4', '8J7', 'B'), ('O0VUFB', '8J7', 'F'), ('SL2EWC',	[('2RZ9YC', 'HLNOAKSSJD95T45', [('KGC', '8PFBI1NC', 'S'), ('L74', '05NTPDLD', 'B'), ('QRH', '3NAQXDK5', 'S'), ('YKQ', 'W8QAL4VU', 'F')]), ('34Q54J', 'QT2TI1266JT1GLS', [('2C0', 'COGOJ1U0', 'B'), ('QRH', '3NAQXDK5', 'C'), ('YKQ', 'W8QAL4VU', 'C')]), ('3HUBU4', '4SNO1JO1OAEB2ZZ', [('2C0', 'COGOJ1U0', 'F'), ('8J7', 'WQQYDXLU', 'B'), ('HV3', 'MP7SXFUU', 'B'), ('KGC', '8PFBI1NC', 'F'), ('L74', '05NTPDLD', 'S'), ('QRH', '3NAQXDK5', 'C')]), ('9JVGIY', 'ETEAXDPZIXKYO97', [('2C0', 'COGOJ1U0', 'B'), ('L2D', 'BR9EZ019', 'S'), ('QRH', '3NAQXDK5',	[('2RZ9YC', 'HLNOAKSSJD95T45', [('KGC', '8PFBI1NC', 'S'), ('L74', '05NTPDLD', 'B'), ('QRH', '3NAQXDK5', 'S'), ('YKQ', 'W8QAL4VU', 'F')]), ('34Q54J', 'QT2TI1266JT1GLS', [('2C0', 'COGOJ1U0', 'B'), ('QRH', '3NAQXDK5', 'C'), ('YKQ', 'W8QAL4VU', 'C')]), ('3HUBU4', '4SNO1JO1OAEB2ZZ', [('2C0', 'COGOJ1U0', 'F'), ('8J7', 'WQQYDXLU', 'B'), ('HV3', 'MP7SXFUU', 'B'), ('KGC', '8PFBI1NC', 'F'), ('L74', '05NTPDLD', 'S'), ('QRH', '3NAQXDK5', 'C')]), ('9JVGIY', 'ETEAXDPZIXKYO97', [('2C0', 'COGOJ1U0', 'B'), ('L2D', 'BR9EZ019', 'S'), ('QRH', '3NAQXDK5',	Passed

```
'L2D', 'D'),
('GX8M2G', 'L2D',
'S'), ('9JVGIY',
'L2D', 'S'),
('Y89DXJ', 'L2D',
'F'), ('O0VUFB',
'L2D', 'A'),
('2RZ9YC', 'YKQ',
'F'), ('T1GNFU',
'YKQ', 'A'),
('GX8M2G', 'YKQ',
'C'), ('O0VUFB',
'YKQ', 'B'),
('34Q54J', 'YKQ',
'C'), ('2RZ9YC',
'L74', 'B'),
('SL2EWC', 'L74',
'A'), ('GX8M2G',
'L74', 'C'),
('3HUBU4', 'L74',
'S'), ('O0VUFB',
'L74', 'A'),
('2RZ9YC', 'QRH',
'S'), ('3HUBU4',
'QRH', 'C'),
('9JVGIY', 'QRH',
'A'), ('O0VUFB',
'QRH', 'S'),
('34Q54J', 'QRH',
'C'), ('2RZ9YC',
'KGC', 'S'),
('C0BF8X', 'KGC',
'S'), ('SL2EWC',
'KGC', 'D'),
('GX8M2G', 'KGC',
'C'), ('3HUBU4',
'KGC', 'F'),
('T1GNFU', '2C0',
'S'), ('C0BF8X',
'2C0', 'S'),
('SL2EWC', '2C0',
'B'), ('GX8M2G',
'2C0', 'B'),
('3HUBU4', '2C0',
'F'), ('9JVGIY',
'2C0', 'B'),
('Y89DXJ', '2C0',
'S'), ('O0VUFB',
'2C0', 'C'),
('34Q54J', '2C0',
'B')])
```

```
'A')]), ('C0BF8X',
'CR4S21F6YAMG3RQ',
[('2C0',
'COGOJ1U0', 'S'),
('KGC',
'8PFBI1NC',
'S')]), ('GX8M2G',
'NUA13O7B1WRWPAM',
[('2C0',
'COGOJ1U0', 'B'),
('8J7',
'WQQYDXLU', 'B'),
('HV3',
'MP7SXFUU', 'S'),
('KGC',
'8PFBI1NC', 'C'),
('L2D',
'BR9EZ019', 'S'),
('L74',
'05NTPDLD', 'C'),
('YKQ',
'W8QAL4VU',
'C')]), ('O0VUFB',
'2QD2CF2PVQWSOW6',
[('2C0',
'COGOJ1U0', 'C'),
('8J7',
'WQQYDXLU', 'F'),
('L2D',
'BR9EZ019', 'A'),
('L74',
'05NTPDLD', 'A'),
('QRH',
'3NAQXDK5', 'S'),
('YKQ',
'W8QAL4VU',
'B')]), ('SL2EWC',
'NQCY3NXRN5KM4GW',
[('2C0',
'COGOJ1U0', 'B'),
('8J7',
'WQQYDXLU', 'F'),
('KGC',
'8PFBI1NC', 'D'),
('L2D',
'BR9EZ019', 'D'),
('L74',
'05NTPDLD',
'A')]), ('T1GNFU',
'X82J3MVZDA7YVL8',
[('2C0',
'COGOJ1U0', 'S'),
('YKQ',
```

```
'A')]), ('C0BF8X',
'CR4S21F6YAMG3RQ',
[('2C0',
'COGOJ1U0', 'S'),
('KGC',
'8PFBI1NC',
'S')]), ('GX8M2G',
'NUA13O7B1WRWPAM',
[('2C0',
'COGOJ1U0', 'B'),
('8J7',
'WQQYDXLU', 'B'),
('HV3',
'MP7SXFUU', 'S'),
('KGC',
'8PFBI1NC', 'C'),
('L2D',
'BR9EZ019', 'S'),
('L74',
'05NTPDLD', 'C'),
('YKQ',
'W8QAL4VU',
'C')]), ('O0VUFB',
'2QD2CF2PVQWSOW6',
[('2C0',
'COGOJ1U0', 'C'),
('8J7',
'WQQYDXLU', 'F'),
('L2D',
'BR9EZ019', 'A'),
('L74',
'05NTPDLD', 'A'),
('QRH',
'3NAQXDK5', 'S'),
('YKQ',
'W8QAL4VU',
'B')]), ('SL2EWC',
'NQCY3NXRN5KM4GW',
[('2C0',
'COGOJ1U0', 'B'),
('8J7',
'WQQYDXLU', 'F'),
('KGC',
'8PFBI1NC', 'D'),
('L2D',
'BR9EZ019', 'D'),
('L74',
'05NTPDLD',
'A')]), ('T1GNFU',
'X82J3MVZDA7YVL8',
[('2C0',
'COGOJ1U0', 'S'),
('YKQ',
```

		'W8QAL4VU', 'A')]), ('Y89DXJ', '0TKLC1UUDETFIL0', [('2C0', 'COGOJ1U0', 'S'), ('HV3', 'MP7SXFUU', 'F'), ('L2D', 'BR9EZ019', 'F')])]\n	'W8QAL4VU', 'A')]), ('Y89DXJ', '0TKLC1UUDETFIL0', [('2C0', 'COGOJ1U0', 'S'), ('HV3', 'MP7SXFUU', 'F'), ('L2D', 'BR9EZ019', 'F')])]\n	
Test Case 7	transcript([('FEB', 'RZPP633E'), ('Q9O', 'VQ5HPC70'), ('ZO1', 'R6I5RWLL'), ('9SN', '9V2EAX9K'), ('65P', 'NXY5TCHZ')], [('3CGLQZ', 'SS6AOTSXM96P52P'), ('WMYEGK', 'JBU3VDGQP60RE1A'), ('NKMJZ5', '9KRKD6PM7J19TRJ'), ('GZRVRF', 'R7N8WO7TBYQ23X9'), ('9OTEHO', 'AQS01Q2Y3474NV7'), ('W89X3M', 'ACFDP5YFPAR6LF5'), ('O4Z3KK', 'U4JB3JCU0HRGETI'), ('61E7B3', 'NAIJ3H459DGK5IL'), ('P4V5ZU', 'SP4KH35P53FHXIX'), ('MVX6YS', '9QCGGSZ3ULZPS12'), ('KC8G8T', 'SPRI3LJFAWQYD4A'), ('RHCFD8', 'MQCEVS2YXLLUE0Q'), ('TNIYJ9', 'ER328QRK0H2O6C0'), ('36O4UQ', 'G99I3OXG4CZKWM4'), ('8LIEZ9', 'CKJBA5ZVWDJLUQF'), ('Q4R1ES', 'E88QAND6OVB562U'), ('643B1W', 'JFNVR1PWAUEX2H3')], [('3CGLQZ', 'FEB', 'S'), ('NKMJZ5',	[('36O4UQ', 'G99I3OXG4CZKWM4', [('65P', 'NXY5TCHZ', 'F'), ('FEB', 'RZPP633E', 'F'), ('Q9O', 'VQ5HPC70', 'A')]), ('3CGLQZ', 'SS6AOTSXM96P52P', [('65P', 'NXY5TCHZ', 'S'), ('9SN', '9V2EAX9K', 'F'), ('FEB', 'RZPP633E', 'S'), ('Q9O', 'VQ5HPC70', 'S')]), ('61E7B3', 'NAIJ3H459DGK5IL', [('65P', 'NXY5TCHZ', 'F'), ('9SN', '9V2EAX9K', 'F'), ('FEB', 'RZPP633E', 'A'), ('ZO1', 'R6I5RWLL', 'C')]), ('643B1W', 'JFNVR1PWAUEX2H3', [('65P', 'NXY5TCHZ', 'B'), ('FEB', 'RZPP633E', 'S'), ('Q9O', 'VQ5HPC70', 'B')]), ('8LIEZ9', 'CKJBA5ZVWDJLUQF', [('9SN', '9V2EAX9K', 'C'), ('FEB', 'RZPP633E', 'A'),	[('36O4UQ', 'G99I3OXG4CZKWM4', [('65P', 'NXY5TCHZ', 'F'), ('FEB', 'RZPP633E', 'F'), ('Q9O', 'VQ5HPC70', 'A')]), ('3CGLQZ', 'SS6AOTSXM96P52P', [('65P', 'NXY5TCHZ', 'S'), ('9SN', '9V2EAX9K', 'F'), ('FEB', 'RZPP633E', 'S'), ('Q9O', 'VQ5HPC70', 'S')]), ('61E7B3', 'NAIJ3H459DGK5IL', [('65P', 'NXY5TCHZ', 'F'), ('9SN', '9V2EAX9K', 'F'), ('FEB', 'RZPP633E', 'A'), ('ZO1', 'R6I5RWLL', 'C')]), ('643B1W', 'JFNVR1PWAUEX2H3', [('65P', 'NXY5TCHZ', 'B'), ('FEB', 'RZPP633E', 'S'), ('Q9O', 'VQ5HPC70', 'B')]), ('8LIEZ9', 'CKJBA5ZVWDJLUQF', [('9SN', '9V2EAX9K', 'C'), ('FEB', 'RZPP633E', 'A'),	Passed

'FEB', 'C'), ('GZRVRF', 'FEB', 'B'), ('W89X3M', 'FEB', 'F'), ('61E7B3', 'FEB', 'A'), ('P4V5ZU', 'FEB', 'F'), ('RHCFD8', 'FEB', 'B'), ('TNIYJ9', 'FEB', 'S'), ('36O4UQ', 'FEB', 'F'), ('8LIEZ9', 'FEB', 'A'), ('643B1W', 'FEB', 'S'), ('3CGLQZ', 'Q9O', 'S'), ('WMYEGK', 'Q9O', 'A'), ('NKMJZ5', 'Q9O', 'C'), ('9OTEHO', 'Q9O', 'F'), ('MVX6YS', 'Q9O', 'C'), ('KC8G8T', 'Q9O', 'S'), ('36O4UQ', 'Q9O', 'A'), ('8LIEZ9', 'Q9O', 'C'), ('Q4R1ES', 'Q9O', 'A'), ('643B1W', 'Q9O', 'B'), ('WMYEGK', 'ZO1', 'B'), ('NKMJZ5', 'ZO1', 'S'), ('9OTEHO', 'ZO1', 'B'), ('O4Z3KK', 'ZO1', 'D'), ('61E7B3', 'ZO1', 'C'), ('P4V5ZU', 'ZO1', 'D'), ('MVX6YS', 'ZO1', 'F'), ('KC8G8T', 'ZO1', 'S'), ('RHCFD8', 'ZO1', 'A'), ('TNIYJ9', 'ZO1', 'C'), ('8LIEZ9', 'ZO1', 'A'), ('3CGLQZ', '9SN', 'F'), ('O4Z3KK', '9SN', 'S'), ('61E7B3', '9SN', 'F'), ('P4V5ZU', '9SN', 'S'), ('8LIEZ9', '9SN',	('Q9O', 'VQ5HPC70', 'C'), ('ZO1', 'R6I5RWLL', 'A')]), ('9OTEHO', 'AQS01Q2Y3474NV7', [('Q9O', 'VQ5HPC70', 'F'), ('ZO1', 'R6I5RWLL', 'B')]), ('GZRVRF', 'R7N8WO7TBYQ23X9', [('65P', 'NXY5TCHZ', 'B'), ('FEB', 'RZPP633E', 'B')]), ('KC8G8T', 'SPRI3LJFAWQYD4A', [('65P', 'NXY5TCHZ', 'D'), ('Q9O', 'VQ5HPC70', 'S'), ('ZO1', 'R6I5RWLL', 'S')]), ('MVX6YS', '9QCGGSZ3ULZPS12', [('Q9O', 'VQ5HPC70', 'C'), ('ZO1', 'R6I5RWLL', 'F')]), ('NKMJZ5', '9KRKD6PM7J19TRJ', [('FEB', 'RZPP633E', 'C'), ('Q9O', 'VQ5HPC70', 'C'), ('ZO1', 'R6I5RWLL', 'S')]), ('O4Z3KK', 'U4JB3JCU0HRGETI', [('9SN', '9V2EAX9K', 'S'), ('ZO1', 'R6I5RWLL', 'D')]), ('P4V5ZU', 'SP4KH35P53FHXIX', [('9SN', '9V2EAX9K', 'S'), ('FEB', 'RZPP633E', 'F'), ('ZO1', 'R6I5RWLL', 'D')]), ('Q4R1ES',	('Q9O', 'VQ5HPC70', 'C'), ('ZO1', 'R6I5RWLL', 'A')]), ('9OTEHO', 'AQS01Q2Y3474NV7', [('Q9O', 'VQ5HPC70', 'F'), ('ZO1', 'R6I5RWLL', 'B')]), ('GZRVRF', 'R7N8WO7TBYQ23X9', [('65P', 'NXY5TCHZ', 'B'), ('FEB', 'RZPP633E', 'B')]), ('KC8G8T', 'SPRI3LJFAWQYD4A', [('65P', 'NXY5TCHZ', 'D'), ('Q9O', 'VQ5HPC70', 'S'), ('ZO1', 'R6I5RWLL', 'S')]), ('MVX6YS', '9QCGGSZ3ULZPS12', [('Q9O', 'VQ5HPC70', 'C'), ('ZO1', 'R6I5RWLL', 'F')]), ('NKMJZ5', '9KRKD6PM7J19TRJ', [('FEB', 'RZPP633E', 'C'), ('Q9O', 'VQ5HPC70', 'C'), ('ZO1', 'R6I5RWLL', 'S')]), ('O4Z3KK', 'U4JB3JCU0HRGETI', [('9SN', '9V2EAX9K', 'S'), ('ZO1', 'R6I5RWLL', 'D')]), ('P4V5ZU', 'SP4KH35P53FHXIX', [('9SN', '9V2EAX9K', 'S'), ('FEB', 'RZPP633E', 'F'), ('ZO1', 'R6I5RWLL', 'D')]), ('Q4R1ES',

	'C'), ('3CGLQZ', '65P', 'S'), ('GZRVRF', '65P', 'B'), ('W89X3M', '65P', 'B'), ('61E7B3', '65P', 'F'), ('KC8G8T', '65P', 'D'), ('TNIYJ9', '65P', 'A'), ('36O4UQ', '65P', 'F'), ('Q4R1ES', '65P', 'S'), ('643B1W', '65P', 'B')])	'E88QAND6OVB562U', [('65P', 'NXY5TCHZ', 'S'), ('Q9O', 'VQ5HPC70', 'A')]), ('RHCFD8', 'MQCEVS2YXLLUE0Q', [('FEB', 'RZPP633E', 'B'), ('ZO1', 'R6I5RWLL', 'A')]), ('TNIYJ9', 'ER328QRK0H2O6C0', [('65P', 'NXY5TCHZ', 'A'), ('FEB', 'RZPP633E', 'S'), ('ZO1', 'R6I5RWLL', 'C')]), ('W89X3M', 'ACFDP5YFPAR6LF5', [('65P', 'NXY5TCHZ', 'B'), ('FEB', 'RZPP633E', 'F')]), ('WMYEGK', 'JBU3VDGQP60RE1A', [('Q9O', 'VQ5HPC70', 'A'), ('ZO1', 'R6I5RWLL', 'B')])]\n	'E88QAND6OVB562U', [('65P', 'NXY5TCHZ', 'S'), ('Q9O', 'VQ5HPC70', 'A')]), ('RHCFD8', 'MQCEVS2YXLLUE0Q', [('FEB', 'RZPP633E', 'B'), ('ZO1', 'R6I5RWLL', 'A')]), ('TNIYJ9', 'ER328QRK0H2O6C0', [('65P', 'NXY5TCHZ', 'A'), ('FEB', 'RZPP633E', 'S'), ('ZO1', 'R6I5RWLL', 'C')]), ('W89X3M', 'ACFDP5YFPAR6LF5', [('65P', 'NXY5TCHZ', 'B'), ('FEB', 'RZPP633E', 'F')]), ('WMYEGK', 'JBU3VDGQP60RE1A', [('Q9O', 'VQ5HPC70', 'A'), ('ZO1', 'R6I5RWLL', 'B')])]\n	
Test Case 8	transcript([('L0H', 'N4AAMFRT'), ('4A3', '7O7QHCJK'), ('KRQ', '62KDZBB1'), ('85U', '6T3OEPVJ'), ('LIX', 'YT128TDH'), ('HCJ', 'E5X8HA27'), ('D0B', 'V8KJG82P'), ('IPP', 'BAC24UO7'), ('E09', 'ODZLFKP5')], [('DG4GIP', 'BFA6YC6MKADKLQT'), ('G2S3FX', 'WPIOG8EQJGAYO8Y'), ('EVCIIF', 'MUPI6WWW3P00E1P'), ('ZTLLZM', 'EA68PCHQD4VU5HU'), ('E1ABJN', '0B8OGE3UVD1VL3T'),	[('ADO2JD', 'QFQ1AYJT70QLTO7', [('4A3', '7O7QHCJK', 'C'), ('E09', 'ODZLFKP5', 'F'), ('HCJ', 'E5X8HA27', 'F'), ('KRQ', '62KDZBB1', 'B'), ('LIX', 'YT128TDH', 'D')]), ('B0ZCI9', 'NMLUKCILLG64CFP', [('4A3', '7O7QHCJK', 'A'), ('E09', 'ODZLFKP5', 'D'), ('HCJ', 'E5X8HA27', 'C'),	[('ADO2JD', 'QFQ1AYJT70QLTO7', [('4A3', '7O7QHCJK', 'C'), ('E09', 'ODZLFKP5', 'F'), ('HCJ', 'E5X8HA27', 'F'), ('KRQ', '62KDZBB1', 'B'), ('LIX', 'YT128TDH', 'D')]), ('B0ZCI9', 'NMLUKCILLG64CFP', [('4A3', '7O7QHCJK', 'A'), ('E09', 'ODZLFKP5', 'D'), ('HCJ', 'E5X8HA27', 'C'),	Passed

	('DMA9F8', 'BEO1IFFN4FMJRKJ'), ('WPL47Q', 'FSN1NHE2Y2EGOVM'), ('ETS14O', 'LR8JWABCO0QGOGH'), ('CUQ58K', 'PIRL1XFYU6UB0TX'), ('B0ZCI9', 'NMLUKCILLG64CFP'), ('ADO2JD', 'QFQ1AYJT70QLTO7')], [('DG4GIP', 'L0H', 'A'), ('G2S3FX', 'L0H', 'D'), ('EVCIIF', 'L0H', 'C'), ('ZTLLZM', 'L0H', 'C'), ('ETS14O', 'L0H', 'C'), ('B0ZCI9', 'L0H', 'B'), ('DG4GIP', '4A3', 'S'), ('EVCIIF', '4A3', 'B'), ('ZTLLZM', '4A3', 'F'), ('WPL47Q', '4A3', 'A'), ('CUQ58K', '4A3', 'C'), ('B0ZCI9', '4A3', 'A'), ('ADO2JD', '4A3', 'C'), ('DG4GIP', 'KRQ', 'C'), ('G2S3FX', 'KRQ', 'C'), ('EVCIIF', 'KRQ', 'A'), ('DMA9F8', 'KRQ', 'A'), ('WPL47Q', 'KRQ', 'B'), ('B0ZCI9', 'KRQ', 'C'), ('ADO2JD', 'KRQ', 'B'), ('DG4GIP', '85U', 'S'), ('G2S3FX', '85U', 'B'), ('EVCIIF', '85U', 'A'), ('ZTLLZM', '85U', 'C'), ('E1ABJN', '85U', 'S'), ('DMA9F8', '85U', 'F'), ('WPL47Q', '85U', 'S'), ('ETS14O',	('KRQ', '62KDZBB1', 'C'), ('L0H', 'N4AAMFRT', 'B')]), ('CUQ58K', 'PIRL1XFYU6UB0TX', [('4A3', '7O7QHCJK', 'C'), ('85U', '6T3OEPVJ', 'A'), ('D0B', 'V8KJG82P', 'A'), ('E09', 'ODZLFKP5', 'D'), ('HCJ', 'E5X8HA27', 'D'), ('IPP', 'BAC24UO7', 'A'), ('LIX', 'YT128TDH', 'F')]), ('DG4GIP', 'BFA6YC6MKADKLQT', [('4A3', '7O7QHCJK', 'S'), ('85U', '6T3OEPVJ', 'S'), ('IPP', 'BAC24UO7', 'D'), ('KRQ', '62KDZBB1', 'C'), ('L0H', 'N4AAMFRT', 'A')]), ('DMA9F8', 'BEO1IFFN4FMJRKJ', [('85U', '6T3OEPVJ', 'F'), ('KRQ', '62KDZBB1', 'A')]), ('E1ABJN', '0B8OGE3UVD1VL3T', [('85U', '6T3OEPVJ', 'S'), ('E09', 'ODZLFKP5', 'C'), ('LIX', 'YT128TDH', 'S')]), ('ETS14O', 'LR8JWABCO0QGOGH', [('85U', '6T3OEPVJ', 'C'), ('E09', 'ODZLFKP5', 'S'), ('HCJ',	('KRQ', '62KDZBB1', 'C'), ('L0H', 'N4AAMFRT', 'B')]), ('CUQ58K', 'PIRL1XFYU6UB0TX', [('4A3', '7O7QHCJK', 'C'), ('85U', '6T3OEPVJ', 'A'), ('D0B', 'V8KJG82P', 'A'), ('E09', 'ODZLFKP5', 'D'), ('HCJ', 'E5X8HA27', 'D'), ('IPP', 'BAC24UO7', 'A'), ('LIX', 'YT128TDH', 'F')]), ('DG4GIP', 'BFA6YC6MKADKLQT', [('4A3', '7O7QHCJK', 'S'), ('85U', '6T3OEPVJ', 'S'), ('IPP', 'BAC24UO7', 'D'), ('KRQ', '62KDZBB1', 'C'), ('L0H', 'N4AAMFRT', 'A')]), ('DMA9F8', 'BEO1IFFN4FMJRKJ', [('85U', '6T3OEPVJ', 'F'), ('KRQ', '62KDZBB1', 'A')]), ('E1ABJN', '0B8OGE3UVD1VL3T', [('85U', '6T3OEPVJ', 'S'), ('E09', 'ODZLFKP5', 'C'), ('LIX', 'YT128TDH', 'S')]), ('ETS14O', 'LR8JWABCO0QGOGH', [('85U', '6T3OEPVJ', 'C'), ('E09', 'ODZLFKP5', 'S'), ('HCJ',

'85U', 'C'), ('CUQ58K', '85U', 'A'), ('G2S3FX', 'LIX', 'C'), ('E1ABJN', 'LIX', 'S'), ('CUQ58K', 'LIX', 'F'), ('ADO2JD', 'LIX', 'D'), ('EVCIIF', 'HCJ', 'A'), ('ZTLLZM', 'HCJ', 'D'), ('WPL47Q', 'HCJ', 'C'), ('ETS14O', 'HCJ', 'C'), ('CUQ58K', 'HCJ', 'D'), ('B0ZCI9', 'HCJ', 'C'), ('ADO2JD', 'HCJ', 'F'), ('EVCIIF', 'D0B', 'D'), ('CUQ58K', 'D0B', 'A'), ('DG4GIP', 'IPP', 'D'), ('G2S3FX', 'IPP', 'S'), ('EVCIIF', 'IPP', 'B'), ('WPL47Q', 'IPP', 'D'), ('ETS14O', 'IPP', 'C'), ('CUQ58K', 'IPP', 'A'), ('G2S3FX', 'E09', 'A'), ('E1ABJN', 'E09', 'C'), ('WPL47Q', 'E09', 'S'), ('ETS14O', 'E09', 'S'), ('CUQ58K', 'E09', 'D'), ('B0ZCI9', 'E09', 'D'), ('ADO2JD', 'E09', 'F')])	'E5X8HA27', 'C'), ('IPP', 'BAC24UO7', 'C'), ('L0H', 'N4AAMFRT', 'C')]), ('EVCIIF', 'MUPI6WWW3P00E1P', [('4A3', '7O7QHCJK', 'B'), ('85U', '6T3OEPVJ', 'A'), ('D0B', 'V8KJG82P', 'D'), ('HCJ', 'E5X8HA27', 'A'), ('IPP', 'BAC24UO7', 'B'), ('KRQ', '62KDZBB1', 'A'), ('L0H', 'N4AAMFRT', 'C')]), ('G2S3FX', 'WPIOG8EQJGAYO8Y', [('85U', '6T3OEPVJ', 'B'), ('E09', 'ODZLFKP5', 'A'), ('IPP', 'BAC24UO7', 'S'), ('KRQ', '62KDZBB1', 'C'), ('L0H', 'N4AAMFRT', 'D'), ('LIX', 'YT128TDH', 'C')]), ('WPL47Q', 'FSN1NHE2Y2EGOVM', [('4A3', '7O7QHCJK', 'A'), ('85U', '6T3OEPVJ', 'S'), ('E09', 'ODZLFKP5', 'S'), ('HCJ', 'E5X8HA27', 'C'), ('IPP', 'BAC24UO7', 'D'), ('KRQ', '62KDZBB1', 'B')]), ('ZTLLZM', 'EA68PCHQD4VU5HU', [('4A3', '7O7QHCJK', 'F'),	'E5X8HA27', 'C'), ('IPP', 'BAC24UO7', 'C'), ('L0H', 'N4AAMFRT', 'C')]), ('EVCIIF', 'MUPI6WWW3P00E1P', [('4A3', '7O7QHCJK', 'B'), ('85U', '6T3OEPVJ', 'A'), ('D0B', 'V8KJG82P', 'D'), ('HCJ', 'E5X8HA27', 'A'), ('IPP', 'BAC24UO7', 'B'), ('KRQ', '62KDZBB1', 'A'), ('L0H', 'N4AAMFRT', 'C')]), ('G2S3FX', 'WPIOG8EQJGAYO8Y', [('85U', '6T3OEPVJ', 'B'), ('E09', 'ODZLFKP5', 'A'), ('IPP', 'BAC24UO7', 'S'), ('KRQ', '62KDZBB1', 'C'), ('L0H', 'N4AAMFRT', 'D'), ('LIX', 'YT128TDH', 'C')]), ('WPL47Q', 'FSN1NHE2Y2EGOVM', [('4A3', '7O7QHCJK', 'A'), ('85U', '6T3OEPVJ', 'S'), ('E09', 'ODZLFKP5', 'S'), ('HCJ', 'E5X8HA27', 'C'), ('IPP', 'BAC24UO7', 'D'), ('KRQ', '62KDZBB1', 'B')]), ('ZTLLZM', 'EA68PCHQD4VU5HU', [('4A3', '7O7QHCJK', 'F'),

		('85U', '6T3OEPVJ', 'C'), ('HCJ', 'E5X8HA27', 'D'), ('L0H', 'N4AAMFRT', 'C')])]\n	('85U', '6T3OEPVJ', 'C'), ('HCJ', 'E5X8HA27', 'D'), ('L0H', 'N4AAMFRT', 'C')])]\n	
Test Case 9	transcript([('RH8', '5K0F3K93'), ('ISL', 'DN4EUOE2'), ('LGT', '1X0VNZ52'), ('NIN', 'W8391UBG'), ('3C9', 'Y7WW04EH'), ('UKS', 'DP4WAW7D'), ('JOC', 'RHRUPQ0X'), ('5SZ', '7TLVVLE9'), ('LM1', 'LYEVLV9Q'), ('BJ9', 'PFFHKBCN')], [('6D8AHB', 'O33LHE2K9M6KTG4'), ('YYVFJW', 'ZJA92U2DMXEADQE'), ('64VIAN', 'SZDOIVATNIEZU38'), ('XPKCCS', '8KDLZLJ6ICQM5JJ'), ('DFM69E', 'U8O00X9M9OAPT2N'), ('FMS2PO', 'PS08CFDZCJ6LVQT'), ('ZNC7WX', 'B8JPYBIH8KUZ4JB'), ('O6M23V', '74AY5VUFMCCP205'), ('HZDHQN', 'NMAC9O9W9D4TB2L'), ('BWRO0B', 'UACK4SVZZ5HUNWU'), ('ERDJCY', 'A3WCGU8G8VGCNCG'), ('4J840Y', '6AZQDUDKI9QFGXP'), ('608LRD', 'ECAITAIFBT11VUC'), ('7SI0KH', 'THLREP6D3R52ZH2'), ('OVCO6I', '9Y346OSSD13NGKZ'), ('085D1O', 'MEZ13XP7HTL6T1L'), ('KV6BL9', 'VIZKWA2LVURV3UU'),	[('085D1O', 'MEZ13XP7HTL6T1L', [('ISL', 'DN4EUOE2', 'D'), ('LGT', '1X0VNZ52', 'F'), ('LM1', 'LYEVLV9Q', 'S'), ('RH8', '5K0F3K93', 'C')]), ('4J840Y', '6AZQDUDKI9QFGXP', [('BJ9', 'PFFHKBCN', 'F'), ('JOC', 'RHRUPQ0X', 'S'), ('NIN', 'W8391UBG', 'B'), ('UKS', 'DP4WAW7D', 'S')]), ('608LRD', 'ECAITAIFBT11VUC', [('3C9', 'Y7WW04EH', 'F'), ('LM1', 'LYEVLV9Q', 'B')]), ('64VIAN', 'SZDOIVATNIEZU38', [('3C9', 'Y7WW04EH', 'D'), ('5SZ', '7TLVVLE9', 'F'), ('BJ9', 'PFFHKBCN', 'F'), ('JOC', 'RHRUPQ0X', 'B'), ('RH8', '5K0F3K93', 'S'), ('UKS', 'DP4WAW7D', 'S')]), ('6D8AHB', 'O33LHE2K9M6KTG4', [('BJ9', 'PFFHKBCN', 'S'), ('JOC',	[('085D1O', 'MEZ13XP7HTL6T1L', [('ISL', 'DN4EUOE2', 'D'), ('LGT', '1X0VNZ52', 'F'), ('LM1', 'LYEVLV9Q', 'S'), ('RH8', '5K0F3K93', 'C')]), ('4J840Y', '6AZQDUDKI9QFGXP', [('BJ9', 'PFFHKBCN', 'F'), ('JOC', 'RHRUPQ0X', 'S'), ('NIN', 'W8391UBG', 'B'), ('UKS', 'DP4WAW7D', 'S')]), ('608LRD', 'ECAITAIFBT11VUC', [('3C9', 'Y7WW04EH', 'F'), ('LM1', 'LYEVLV9Q', 'B')]), ('64VIAN', 'SZDOIVATNIEZU38', [('3C9', 'Y7WW04EH', 'D'), ('5SZ', '7TLVVLE9', 'F'), ('BJ9', 'PFFHKBCN', 'F'), ('JOC', 'RHRUPQ0X', 'B'), ('RH8', '5K0F3K93', 'S'), ('UKS', 'DP4WAW7D', 'S')]), ('6D8AHB', 'O33LHE2K9M6KTG4', [('BJ9', 'PFFHKBCN', 'S'), ('JOC',	Passed

('9O2L2Y', 'HZ9SS7KFTZ7PETL')], [('64VIAN', 'RH8', 'S'), ('XPKCCS', 'RH8', 'B'), ('DFM69E', 'RH8', 'C'), ('ZNC7WX', 'RH8', 'F'), ('HZDHQN', 'RH8', 'A'), ('7SI0KH', 'RH8', 'S'), ('OVCO6I', 'RH8', 'S'), ('085D1O', 'RH8', 'C'), ('KV6BL9', 'RH8', 'F'), ('YYVFJW', 'ISL', 'F'), ('XPKCCS', 'ISL', 'B'), ('FMS2PO', 'ISL', 'D'), ('ZNC7WX', 'ISL', 'B'), ('BWRO0B', 'ISL', 'S'), ('ERDJCY', 'ISL', 'B'), ('OVCO6I', 'ISL', 'A'), ('085D1O', 'ISL', 'D'), ('KV6BL9', 'ISL', 'A'), ('6D8AHB', 'LGT', 'C'), ('XPKCCS', 'LGT', 'C'), ('FMS2PO', 'LGT', 'B'), ('ZNC7WX', 'LGT', 'F'), ('HZDHQN', 'LGT', 'A'), ('ERDJCY', 'LGT', 'S'), ('085D1O', 'LGT', 'F'), ('9O2L2Y', 'LGT', 'F'), ('6D8AHB', 'NIN', 'D'), ('YYVFJW', 'NIN', 'S'), ('XPKCCS', 'NIN', 'F'), ('DFM69E', 'NIN', 'S'), ('FMS2PO', 'NIN', 'C'), ('ZNC7WX', 'NIN', 'F'), ('O6M23V', 'NIN', 'B'), ('HZDHQN', 'NIN', 'A'),	'RHRUPQ0X', 'F'), ('LGT', '1X0VNZ52', 'C'), ('LM1', 'LYEVLV9Q', 'F'), ('NIN', 'W8391UBG', 'D')]), ('7SI0KH', 'THLREP6D3R52ZH2', [('LM1', 'LYEVLV9Q', 'D'), ('NIN', 'W8391UBG', 'C'), ('RH8', '5K0F3K93', 'S'), ('UKS', 'DP4WAW7D', 'D')]), ('9O2L2Y', 'HZ9SS7KFTZ7PETL', [('BJ9', 'PFFHKBCN', 'A'), ('LGT', '1X0VNZ52', 'F'), ('LM1', 'LYEVLV9Q', 'S'), ('UKS', 'DP4WAW7D', 'A')]), ('BWRO0B', 'UACK4SVZZ5HUNWU', [('5SZ', '7TLVVLE9', 'F'), ('ISL', 'DN4EUOE2', 'S'), ('JOC', 'RHRUPQ0X', 'S'), ('LM1', 'LYEVLV9Q', 'F'), ('UKS', 'DP4WAW7D', 'F')]), ('DFM69E', 'U8O00X9M9OAPT2N', [('3C9', 'Y7WW04EH', 'B'), ('BJ9', 'PFFHKBCN', 'A'), ('NIN', 'W8391UBG', 'S'), ('RH8', '5K0F3K93', 'C'), ('UKS', 'DP4WAW7D', 'F')]), ('ERDJCY', 'A3WCGU8G8VGCNCG',	'RHRUPQ0X', 'F'), ('LGT', '1X0VNZ52', 'C'), ('LM1', 'LYEVLV9Q', 'F'), ('NIN', 'W8391UBG', 'D')]), ('7SI0KH', 'THLREP6D3R52ZH2', [('LM1', 'LYEVLV9Q', 'D'), ('NIN', 'W8391UBG', 'C'), ('RH8', '5K0F3K93', 'S'), ('UKS', 'DP4WAW7D', 'D')]), ('9O2L2Y', 'HZ9SS7KFTZ7PETL', [('BJ9', 'PFFHKBCN', 'A'), ('LGT', '1X0VNZ52', 'F'), ('LM1', 'LYEVLV9Q', 'S'), ('UKS', 'DP4WAW7D', 'A')]), ('BWRO0B', 'UACK4SVZZ5HUNWU', [('5SZ', '7TLVVLE9', 'F'), ('ISL', 'DN4EUOE2', 'S'), ('JOC', 'RHRUPQ0X', 'S'), ('LM1', 'LYEVLV9Q', 'F'), ('UKS', 'DP4WAW7D', 'F')]), ('DFM69E', 'U8O00X9M9OAPT2N', [('3C9', 'Y7WW04EH', 'B'), ('BJ9', 'PFFHKBCN', 'A'), ('NIN', 'W8391UBG', 'S'), ('RH8', '5K0F3K93', 'C'), ('UKS', 'DP4WAW7D', 'F')]), ('ERDJCY', 'A3WCGU8G8VGCNCG',

('4J840Y', 'NIN', 'B'), ('7SI0KH', 'NIN', 'C'), ('64VIAN', '3C9', 'D'), ('DFM69E', '3C9', 'B'), ('FMS2PO', '3C9', 'F'), ('ZNC7WX', '3C9', 'B'), ('ERDJCY', '3C9', 'S'), ('608LRD', '3C9', 'F'), ('OVCO6I', '3C9', 'D'), ('YYVFJW', 'UKS', 'B'), ('64VIAN', 'UKS', 'S'), ('XPKCCS', 'UKS', 'D'), ('DFM69E', 'UKS', 'F'), ('ZNC7WX', 'UKS', 'F'), ('O6M23V', 'UKS', 'S'), ('BWRO0B', 'UKS', 'F'), ('ERDJCY', 'UKS', 'B'), ('4J840Y', 'UKS', 'S'), ('7SI0KH', 'UKS', 'D'), ('KV6BL9', 'UKS', 'B'), ('9O2L2Y', 'UKS', 'A'), ('6D8AHB', 'JOC', 'F'), ('64VIAN', 'JOC', 'B'), ('XPKCCS', 'JOC', 'A'), ('FMS2PO', 'JOC', 'S'), ('ZNC7WX', 'JOC', 'B'), ('HZDHQN', 'JOC', 'S'), ('BWRO0B', 'JOC', 'S'), ('ERDJCY', 'JOC', 'S'), ('4J840Y', 'JOC', 'S'), ('OVCO6I', 'JOC', 'F'), ('KV6BL9', 'JOC', 'A'), ('64VIAN', '5SZ', 'F'), ('FMS2PO', '5SZ', 'F'), ('O6M23V', '5SZ', 'A'), ('BWRO0B',	[('3C9', 'Y7WW04EH', 'S'), ('5SZ', '7TLVVLE9', 'B'), ('BJ9', 'PFFHKBCN', 'F'), ('ISL', 'DN4EUOE2', 'B'), ('JOC', 'RHRUPQ0X', 'S'), ('LGT', '1X0VNZ52', 'S'), ('UKS', 'DP4WAW7D', 'B')]), ('FMS2PO', 'PS08CFDZCJ6LVQT', [('3C9', 'Y7WW04EH', 'F'), ('5SZ', '7TLVVLE9', 'F'), ('BJ9', 'PFFHKBCN', 'B'), ('ISL', 'DN4EUOE2', 'D'), ('JOC', 'RHRUPQ0X', 'S'), ('LGT', '1X0VNZ52', 'B'), ('NIN', 'W8391UBG', 'C')]), ('HZDHQN', 'NMAC9O9W9D4TB2L', [('JOC', 'RHRUPQ0X', 'S'), ('LGT', '1X0VNZ52', 'A'), ('NIN', 'W8391UBG', 'A'), ('RH8', '5K0F3K93', 'A')]), ('KV6BL9', 'VIZKWA2LVURV3UU', [('5SZ', '7TLVVLE9', 'A'), ('BJ9', 'PFFHKBCN', 'F'), ('ISL', 'DN4EUOE2', 'A'), ('JOC', 'RHRUPQ0X', 'A'), ('RH8', '5K0F3K93', 'F'), ('UKS',	[('3C9', 'Y7WW04EH', 'S'), ('5SZ', '7TLVVLE9', 'B'), ('BJ9', 'PFFHKBCN', 'F'), ('ISL', 'DN4EUOE2', 'B'), ('JOC', 'RHRUPQ0X', 'S'), ('LGT', '1X0VNZ52', 'S'), ('UKS', 'DP4WAW7D', 'B')]), ('FMS2PO', 'PS08CFDZCJ6LVQT', [('3C9', 'Y7WW04EH', 'F'), ('5SZ', '7TLVVLE9', 'F'), ('BJ9', 'PFFHKBCN', 'B'), ('ISL', 'DN4EUOE2', 'D'), ('JOC', 'RHRUPQ0X', 'S'), ('LGT', '1X0VNZ52', 'B'), ('NIN', 'W8391UBG', 'C')]), ('HZDHQN', 'NMAC9O9W9D4TB2L', [('JOC', 'RHRUPQ0X', 'S'), ('LGT', '1X0VNZ52', 'A'), ('NIN', 'W8391UBG', 'A'), ('RH8', '5K0F3K93', 'A')]), ('KV6BL9', 'VIZKWA2LVURV3UU', [('5SZ', '7TLVVLE9', 'A'), ('BJ9', 'PFFHKBCN', 'F'), ('ISL', 'DN4EUOE2', 'A'), ('JOC', 'RHRUPQ0X', 'A'), ('RH8', '5K0F3K93', 'F'), ('UKS',

```
'5SZ', 'F'),
('ERDJCY', '5SZ',
'B'), ('KV6BL9',
'5SZ', 'A'),
('6D8AHB', 'LM1',
'F'), ('XPKCCS',
'LM1', 'D'),
('ZNC7WX', 'LM1',
'C'), ('O6M23V',
'LM1', 'S'),
('BWRO0B', 'LM1',
'F'), ('608LRD',
'LM1', 'B'),
('7SI0KH', 'LM1',
'D'), ('OVCO6I',
'LM1', 'D'),
('085D1O', 'LM1',
'S'), ('9O2L2Y',
'LM1', 'S'),
('6D8AHB', 'BJ9',
'S'), ('64VIAN',
'BJ9', 'F'),
('XPKCCS', 'BJ9',
'B'), ('DFM69E',
'BJ9', 'A'),
('FMS2PO', 'BJ9',
'B'), ('O6M23V',
'BJ9', 'A'),
('ERDJCY', 'BJ9',
'F'), ('4J840Y',
'BJ9', 'F'),
('OVCO6I', 'BJ9',
'B'), ('KV6BL9',
'BJ9', 'F'),
('9O2L2Y', 'BJ9',
'A')])
```

```
'DP4WAW7D',
'B')]), ('O6M23V',
'74AY5VUFMCCP205',
[('5SZ',
'7TLVVLE9', 'A'),
('BJ9',
'PFFHKBCN', 'A'),
('LM1',
'LYEVLV9Q', 'S'),
('NIN',
'W8391UBG', 'B'),
('UKS',
'DP4WAW7D',
'S')]), ('OVCO6I',
'9Y346OSSD13NGKZ',
[('3C9',
'Y7WW04EH', 'D'),
('BJ9',
'PFFHKBCN', 'B'),
('ISL',
'DN4EUOE2', 'A'),
('JOC',
'RHRUPQ0X', 'F'),
('LM1',
'LYEVLV9Q', 'D'),
('RH8',
'5K0F3K93',
'S')]), ('XPKCCS',
'8KDLZLJ6ICQM5JJ',
[('BJ9',
'PFFHKBCN', 'B'),
('ISL',
'DN4EUOE2', 'B'),
('JOC',
'RHRUPQ0X', 'A'),
('LGT',
'1X0VNZ52', 'C'),
('LM1',
'LYEVLV9Q', 'D'),
('NIN',
'W8391UBG', 'F'),
('RH8',
'5K0F3K93', 'B'),
('UKS',
'DP4WAW7D',
'D')]), ('YYVFJW',
'ZJA92U2DMXEADQE',
[('ISL',
'DN4EUOE2', 'F'),
('NIN',
'W8391UBG', 'S'),
('UKS',
'DP4WAW7D',
```

'DP4WAW7D',
'B')]), ('O6M23V',
'74AY5VUFMCCP205',
[('5SZ',
'7TLVVLE9', 'A'),
('BJ9',
'PFFHKBCN', 'A'),
('LM1',
'LYEVLV9Q', 'S'),
('NIN',
'W8391UBG', 'B'),
('UKS',
'DP4WAW7D',
'S')]), ('OVCO6I',
'9Y346OSSD13NGKZ',
[('3C9',
'Y7WW04EH', 'D'),
('BJ9',
'PFFHKBCN', 'B'),
('ISL',
'DN4EUOE2', 'A'),
('JOC',
'RHRUPQ0X', 'F'),
('LM1',
'LYEVLV9Q', 'D'),
('RH8',
'5K0F3K93',
'S')]), ('XPKCCS',
'8KDLZLJ6ICQM5JJ',
[('BJ9',
'PFFHKBCN', 'B'),
('ISL',
'DN4EUOE2', 'B'),
('JOC',
'RHRUPQ0X', 'A'),
('LGT',
'1X0VNZ52', 'C'),
('LM1',
'LYEVLV9Q', 'D'),
('NIN',
'W8391UBG', 'F'),
('RH8',
'5K0F3K93', 'B'),
('UKS',
'DP4WAW7D',
'D')]), ('YYVFJW',
'ZJA92U2DMXEADQE',
[('ISL',
'DN4EUOE2', 'F'),
('NIN',
'W8391UBG', 'S'),
('UKS',
'DP4WAW7D',

		'B')]), ('ZNC7WX', 'B8JPYBIH8KUZ4JB', [('3C9', 'Y7WW04EH', 'B'), ('ISL', 'DN4EUOE2', 'B'), ('JOC', 'RHRUPQ0X', 'B'), ('LGT', '1X0VNZ52', 'F'), ('LM1', 'LYEVLV9Q', 'C'), ('NIN', 'W8391UBG', 'F'), ('RH8', '5K0F3K93', 'F'), ('UKS', 'DP4WAW7D', 'F')])]\n	'B')]), ('ZNC7WX', 'B8JPYBIH8KUZ4JB', [('3C9', 'Y7WW04EH', 'B'), ('ISL', 'DN4EUOE2', 'B'), ('JOC', 'RHRUPQ0X', 'B'), ('LGT', '1X0VNZ52', 'F'), ('LM1', 'LYEVLV9Q', 'C'), ('NIN', 'W8391UBG', 'F'), ('RH8', '5K0F3K93', 'F'), ('UKS', 'DP4WAW7D', 'F')])]\n	
Test Case 10	transcript([('89V', 'RZ1FPDH1'), ('9CE', '13IEY6UN'), ('7IC', '0J6H887T'), ('0S4', 'OJMQ9A5U')], [('GTL1B6', 'ZSDVA6KHPN74RPJ'), ('X1I5I3', 'KEOQGL8YJ1FUKAK'), ('GCLWUW', 'IYQTRGF238SQQ74'), ('7RZX51', '84D6SK5SU87VLHX'), ('69W611', '1CRZXJEFKXZJV7O'), ('X5RTIG', 'D0MKSDQBHT4LFDS'), ('URTLCO', 'L220QGH3AJXXITW'), ('DIAYUX', 'SF8ZC2T4I4F0YQC'), ('AX0MKW', 'R6UZJ21D3MYMNHP'), ('BZD6QK', 'RSZ9ZJFPK4ZQCAG')], [('7RZX51', '89V', 'F'), ('URTLCO', '89V', 'S'), ('DIAYUX', '89V', 'D'), ('X1I5I3', '9CE', 'B'), ('69W611', '9CE', 'A'), ('URTLCO',	[('69W611', '1CRZXJEFKXZJV7O', [('0S4', 'OJMQ9A5U', 'C'), ('7IC', '0J6H887T', 'S'), ('9CE', '13IEY6UN', 'A')]), ('7RZX51', '84D6SK5SU87VLHX', [('89V', 'RZ1FPDH1', 'F')]), ('AX0MKW', 'R6UZJ21D3MYMNHP', [('0S4', 'OJMQ9A5U', 'C'), ('7IC', '0J6H887T', 'D')]), ('BZD6QK', 'RSZ9ZJFPK4ZQCAG', [('0S4', 'OJMQ9A5U', 'C'), ('9CE', '13IEY6UN', 'S')]), ('DIAYUX', 'SF8ZC2T4I4F0YQC', [('0S4', 'OJMQ9A5U', 'B'), ('89V', 'RZ1FPDH1', 'D'), ('9CE', '13IEY6UN', 'F')]), ('GCLWUW',	[('69W611', '1CRZXJEFKXZJV7O', [('0S4', 'OJMQ9A5U', 'C'), ('7IC', '0J6H887T', 'S'), ('9CE', '13IEY6UN', 'A')]), ('7RZX51', '84D6SK5SU87VLHX', [('89V', 'RZ1FPDH1', 'F')]), ('AX0MKW', 'R6UZJ21D3MYMNHP', [('0S4', 'OJMQ9A5U', 'C'), ('7IC', '0J6H887T', 'D')]), ('BZD6QK', 'RSZ9ZJFPK4ZQCAG', [('0S4', 'OJMQ9A5U', 'C'), ('9CE', '13IEY6UN', 'S')]), ('DIAYUX', 'SF8ZC2T4I4F0YQC', [('0S4', 'OJMQ9A5U', 'B'), ('89V', 'RZ1FPDH1', 'D'), ('9CE', '13IEY6UN', 'F')]), ('GCLWUW',	Passed

'9CE', 'F'), ('DIAYUX', '9CE', 'F'), ('BZD6QK', '9CE', 'S'), ('X1I5I3', '7IC', 'S'), ('GCLWUW', '7IC', 'A'), ('69W611', '7IC', 'S'), ('X5RTIG', '7IC', 'D'), ('AX0MKW', '7IC', 'D'), ('GTL1B6', '0S4', 'B'), ('X1I5I3', '0S4', 'F'), ('69W611', '0S4', 'C'), ('DIAYUX', '0S4', 'B'), ('AX0MKW', '0S4', 'C'), ('BZD6QK', '0S4', 'C')])	'IYQTRGF238SQQ74', [('7IC', '0J6H887T', 'A')]), ('GTL1B6', 'ZSDVA6KHPN74RPJ', [('0S4', 'OJMQ9A5U', 'B')]), ('URTLCO', 'L220QGH3AJXXITW', [('89V', 'RZ1FPDH1', 'S'), ('9CE', '13IEY6UN', 'F')]), ('X1I5I3', 'KEOQGL8YJ1FUKAK', [('0S4', 'OJMQ9A5U', 'F'), ('7IC', '0J6H887T', 'S'), ('9CE', '13IEY6UN', 'B')]), ('X5RTIG', 'D0MKSDQBHT4LFDS', [('7IC', '0J6H887T', 'D')])]\n	'IYQTRGF238SQQ74', [('7IC', '0J6H887T', 'A')]), ('GTL1B6', 'ZSDVA6KHPN74RPJ', [('0S4', 'OJMQ9A5U', 'B')]), ('URTLCO', 'L220QGH3AJXXITW', [('89V', 'RZ1FPDH1', 'S'), ('9CE', '13IEY6UN', 'F')]), ('X1I5I3', 'KEOQGL8YJ1FUKAK', [('0S4', 'OJMQ9A5U', 'F'), ('7IC', '0J6H887T', 'S'), ('9CE', '13IEY6UN', 'B')]), ('X5RTIG', 'D0MKSDQBHT4LFDS', [('7IC', '0J6H887T', 'D')])]\n

The due date for submitting this assignment has passed.

10 out of 10 tests passed.

You scored 100.0/100.

Assignment submitted on 2021-11-01, 13:33 IST

Your last recorded submission was :

```python
def histogram(l):
    count = []
    unique = []
    for i in l:
        if i not in unique:
            unique.append(i)
            count.append(l.count(i))
    pair = []
    for i in range(len(count)):
        pair.append((unique[i],count[i]))

    for i in range(len(pair)-1):
        for j in range(i+1,len(pair)):
            if  (pair[i][1])>(pair[j][1]):
                pair[i],pair[j]=pair[j],pair[i]
            elif (pair[i][1]) == (pair[j][1]):
                if (pair[i][0]) > (pair[j][0]):
                    pair[i],pair[j]=pair[j],pair[i]
            else:
                pair[i],pair[j] = pair[i],pair[j]

    return pair

def transcript (coursedetails,studentdetails,grades):
    consolidated =[]
    for i in studentdetails:
        new = []
        for j in grades:
            if i[0]==j[0]:
                for k in coursedetails:
                    if j[1]==k[0]:
                        new.append((j[1],k[1],j[2]))
                        new.sort()
        consolidated.append((i[0],i[1],new))
        consolidated.sort()
    return consolidated
import ast

def tolist(inp):
  inp = "["+inp+"]"
  inp = ast.literal_eval(inp)
  return (inp[0],inp[1])

def parse(inp):
  inp = ast.literal_eval(inp)
  return (inp)

fncall = input()
lparen = fncall.find("(")
rparen = fncall.rfind(")")
fname = fncall[:lparen]
farg = fncall[lparen+1:rparen]

if fname == "histogram":
   arg = ast.literal_eval(farg)
   print(histogram(arg),end="\n")
elif fname == "transcript":
   arg = ast.literal_eval(farg)
   print(transcript(arg[0],arg[1],arg[2]),end="\n")
else:
   print("Function", fname, "unknown")


```

Sample solutions (Provided by instructor)

```
#-----------------------------------

def build_table(l):
    # Use a dictionary to build a frequency table
    frequency = {}

    # For each number, create a new entry in the table or increment the frequer
    for n in l:
        if n in frequency.keys():
            frequency[n] = frequency[n] + 1
        else:
            frequency[n] = 1

    return(frequency)

def sort_table(fdict):
    # First build a list of the form (r,n)
    flist = []
    for n in fdict.keys():
        flist.append((fdict[n],n))
    # Sort this list using built in sort, which will sort first by frequency, t
    flist.sort()

    # Flip each pair and return
    revflist = []
    for (r,n) in flist:
        revflist.append((n,r))
    return(revflist)

def histogram(l):
    frequency_table = build_table(l)
    return(sort_table(frequency_table))

#-----------------------------------

def transcript(coursedetails,studentdetails,grades):

    coursedict = setup_coursedict(coursedetails)
    studentdict = setup_studentdict(studentdetails)
    gradedict = setup_gradedict(grades)

```

```python

    outputlist = []

    for r in sorted(gradedict.keys()):
        gradelist = []
        for (ccode,grade) in sorted(gradedict[r]):
            gradelist.append((ccode,coursedict[ccode],grade))

        outputlist.append((r,studentdict[r],gradelist))

    return(outputlist)

def setup_coursedict(details):

    dict = {}
    for (ccode,cname) in details:
        dict[ccode] = cname

    return(dict)

def setup_studentdict(details):

    dict = {}
    for (rollno,name) in details:
        dict[rollno] = name

    return(dict)

def setup_gradedict(details):

    dict = {}
    for (rollno,ccode,grade) in details:
        if rollno in dict.keys():
            dict[rollno].append((ccode,grade))
        else:
            dict[rollno] = [(ccode,grade)]

    return(dict)

#---------------------------------

import ast

def tolist(inp):
  inp = "["+inp+"]"
  inp = ast.literal_eval(inp)
  return (inp[0],inp[1])

def parse(inp):
  inp = ast.literal_eval(inp)
  return (inp)

fncall = input()
lparen = fncall.find("(")
rparen = fncall.rfind(")")
fname = fncall[:lparen]
farg = fncall[lparen+1:rparen]

if fname == "histogram":
   arg = ast.literal_eval(farg)
   print(histogram(arg),end="\n")
elif fname == "transcript":
   arg = ast.literal_eval(farg)
   print(transcript(arg[0],arg[1],arg[2]),end="\n")
else:
   print("Function", fname, "unknown")


```