

ISE4 Mini Project

Group 19

Book Scanning

Team Members:

1. Aditi Bhosale (2019140009)
2. Anushka Agarwal (2019140002)
3. Shweta Chavan (2019140011)

Project Code:

```
import numpy as np
import operator
import sys

# signup score and library. We have to sort based on signup score, and hence item
getter(0) since based on array1 we are sorting
def parallel_sort(array_1, array_2, ascending=False):
    if not ascending:
        return zip(*sorted(zip(array_1, array_2), key=operator.itemgetter(0))[:, -1])
    return zip(*sorted(zip(array_1, array_2), key=operator.itemgetter(0)))
# zip is to create two arrays such that their keys are matching.

class Library:

    def __init__(self, id, book_ids, signup_days, max_books_scanned_per_day):
        self.id = id
        self.book_ids = book_ids
        self.signup_days = signup_days
        self.max_books_scanned_per_day = max_books_scanned_per_day

    def get_best_book_ids(self, start_day=0):
        global book_scores, libraries_signup_days,
libraries_max_books_scanned_per_day, D, final_books
        available_days = D - self.signup_days - start_day
        #start day is different for different libraries, and available days also
depends on previous libraries which had been there before.

        available_books = list(set(self.book_ids) - set(final_books))
        current_book_scores = np.take(book_scores, available_books)
        # returns array of same type with all the values of given indices. So here
np.take will return an array which contains
        # book scores of all available books
```

[illegible]

```
for x in range(len(book_scores)):
    print(book_scores[x], end=" ")
print("")
pos = 1
# library attributes : hence size of all below parameters are L
libraries_num_books = np.zeros(L) # number of books in lib 0,1,2 etc hence array
of size L
libraries_signup_days = np.zeros(L) # signup days for the libraries hence array
of size L
libraries_max_books_scanned_per_day = np.zeros(L) # max books scanned per day for
the libraries

libraries = np.empty(L, dtype=Library) # array number of elements in row is L
and data type is Library
for i in range(L):
    print("\n\n\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t-***** Library", i, "*****--",
          "\n\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\tBooks \\t Signup days \\t Books shipped per
day ")
    pos += 1
    n, t, m = list(map(int, content[pos].split(' ')))
    libraries_num_books[i] = n
    libraries_signup_days[i] = t
    libraries_max_books_scanned_per_day[i] = m
    print("\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t", n, "\\t\\t\\t", t, "\\t\\t\\t\\t\\t", m)
    pos += 1
    print("\\n\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\tEnter Book ids :", end='')

    book_ids = np.asarray(list(map(int, content[pos].split(' '))))
    for x in range(len(book_ids)):
        print("\\t", book_ids[x], end=" ")

    libraries[i] = Library(i, book_ids, t, m)

# Vectorization is used to speed up the Python code without using loop.
# Using such a function can help in minimizing the running time of code
efficiently.
library_book_score_counter = np.vectorize(lambda library:
library.get_best_books_score())
libraries_scores = library_book_score_counter(libraries)
# library_book_score_counter counts the score of the libraries,
# we pass the array libraries and vectorize takes in each element of the array passed
and gets the best book score
# for it.The result is stored in libraries_scores
heuristic_score = np.vectorize(lambda book_score, signup_days: book_score /
signup_days)
signup_scores = heuristic_score(libraries_scores, libraries_signup_days)

signup_scores, libraries_sorted = parallel_sort(signup_scores, libraries)

final_books = set()
with open(
    "outputs/" + file + ".out", 'w+') as f:

    f.write(str(L) + "\\n")
    print()
    for z in range(60):
        print("*-*", end=' ')

    print('\\n\\n\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\t\\tNumber of libraries scanned up for signing
= ', str(L))
```

[illegible]

Project Output:

Test Case 1:

a_example

```
*****
Enter the number of Books,Libraries,Days:
6 2 7
Enter the Book scores: 1 2 3 6 5 4

***** Library 0 *****
Books   Signup days   Books shipped per day
5       2             2

Enter Book ids :   0 1 2 3 4

***** Library 1 *****
Books   Signup days   Books shipped per day
4       3             1

Enter Book ids :   0 2 3 5
*****

Number of libraries scanned up for signing = 2

Library chosen           Number of books scanned from library
0                         5

Book ID's of books chosen from library : 0 1 2 3 4
*****

Number of libraries scanned up for signing = 2

Library chosen           Number of books scanned from library
0                         5

Book ID's of books chosen from library : 0 1 2 3 4

Library chosen           Number of books scanned from library
1                         1

Book ID's of books chosen from library : 5

Total score: 21

*****
Process finished with exit code 0
|
```

Test Case 2:

```
b_read_on
```

```
Total score: 5822900
```

```
Process finished with exit code 0
```

Test Case 3:

```
c_incunabula
```

```
Total score: 5645747
```

```
Process finished with exit code 0
```

Test Case 4:

```
d_tough_choices
```

```
Total score: 4812730
```

```
Process finished with exit code 0
```

Test Case 5:

```
e_so_many_books
```

```
Total score: 5019670
```

```
Process finished with exit code 0
```

Test Case 6:

```
f_libraries_of_the_world
```

```
Total score: 5240161
```

```
Process finished with exit code 0
```