Dictionaries

Data lookup
Mid-term exam retake

• Retake of mid-term test will be on Tuesday, November 27th, in computer class 321.
  • 14:15 – retake of part 1
  • 15:15 – retake of part 2

• It is possible to retake both parts or either part separately. No formal registration needed.

• The format of the retake is the same as was in the mid-term exam: part 1 is test on paper, part 2 is programming exercise.
Individual consultation

• There will be a possibility for individual consultations, starting from Tuesday, November 27th, around 11:15, in room 322a.

• The topics can cover all aspects of the course, either related or not related to mid-term exam.

• Please reserve a time in Moodle, when next week’s materials are available.
Project

• Project is a larger program that summarizes the concepts learned in the course.
• Work amount approximately 20 hours, deadline Session 15 on December 19th.
• Project formulation should contain the following elements:
  • Vision and purpose of the program.
  • Short description of how the program achieves its objectives.
  • Overview of functions, conditional statements and loops used in the program.
• If you have not yet submitted your project description, please do it today (through Moodle)!
• https://courses.cs.ut.ee/2018/nkp/fall/Main/Project
Dictionaries

Dictionary is a set of key-value pairs.

\[
dict = \{
    \text{key1: value1,}
    \text{key2: value2,}
    \text{key3: value3,}
    \ldots
}\]

\[
dim = \{
    \text{\'length\': 100,}
    \text{\'width\': 50,}
    \text{\'height\': 20}
\}
\]
Adding values

Values can be added using the same syntax as is used for lists.

\[
dict[key] = value
\]

\[
dim['depth'] = 15
\]

Existing values can be changed:

\[
dict[key] = newvalue
\]

\[
dim['depth'] = 25
\]

\[
dim['depth'] -= 10
\]
Quering values

Values can be queried in similar way as values of elements from a list.

```
>>> dict[key1]    >>> dim['length']
value1           100

>>> dict[key2]    >>> dim['depth']
value2           15
```
Checking if key exists in a dictionary

\[
\text{if } key \text{ in } \text{dict:} \quad \text{if 'width' in dim:}
\]

\[
\text{\quad statements} \quad \text{\quad print(dim['width'])}
\]

\[
\text{\quad else:} \quad \text{\quad else:}
\]

\[
\text{\quad \quad print('Not found')} \quad \text{\quad \quad print('Not found')}
\]
Looping over all keys of dictionary

```python
for key in dict:
    statements

for key in dim:
    print('Key name', key)
    print('Key value is', dim[key])
```
Example

Write a program that reads text from a file and prints out the word that occurs with the highest frequency.
Solution

```python
f = open('file.txt')
dict = {}
for row in f:
    parts = row.split()
    for p in parts:
        if not p in dict:
            dict[p] = 1
        else:
            dict[p] += 1
f.close()

maxkey = ""
maxfreq = -1
for key in dict:
    if dict[key] > maxfreq:
        maxkey = key
        maxfreq = dict[key]
print(maxkey, maxfreq)
```
Example 2

Write a program that asks the user for a csv file, keys column number and values column number, and allows the user to perform searches on keys.

Contents of ip.csv:

```plaintext
www.ut.ee,193.40.5.73
www.cs.ut.ee,193.40.5.227
www.ttu.ee,193.40.254.28
www.tlu.ee,193.40.239.30
wikipedia.org,198.35.26.96
postimees.ee,185.154.220.180
twitch.tv,151.101.2.167
eki.ee,193.40.113.42
delfi.lv,62.63.137.114
svt.se,194.15.212.182
```

Keys column 0, values column 1:

- Enter key: wikipedia.org 198.35.26.96
- Enter key: www.ut.ee 193.40.5.73

Keys column 1, values column 0:

- Enter key: 193.40.5.227
- Enter key: www.cs.ut.ee
- Enter key: 999.999.999.99 Not found
Example 2 (cont)

Contents of names.csv:

<table>
<thead>
<tr>
<th>Name1</th>
<th>Name2</th>
<th>Type</th>
<th>SSN</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teet</td>
<td>Alberg</td>
<td>M</td>
<td>B78553</td>
<td>N16</td>
</tr>
<tr>
<td>Karl</td>
<td>Põldsmann</td>
<td>M</td>
<td>B78056</td>
<td>N16</td>
</tr>
<tr>
<td>Geoff</td>
<td>Marvet</td>
<td>M</td>
<td>B87556</td>
<td>N16</td>
</tr>
<tr>
<td>Martina</td>
<td>Sarv</td>
<td>F</td>
<td>B51296</td>
<td>N10</td>
</tr>
<tr>
<td>Karen</td>
<td>Solmaan</td>
<td>F</td>
<td>B87565</td>
<td>N12</td>
</tr>
<tr>
<td>Pille</td>
<td>Jool</td>
<td>F</td>
<td>B87517</td>
<td>N12</td>
</tr>
<tr>
<td>Artjom</td>
<td>Tamm</td>
<td>M</td>
<td>B69805</td>
<td>N12</td>
</tr>
<tr>
<td>Petra</td>
<td>Brit</td>
<td>F</td>
<td>B78908</td>
<td>N10</td>
</tr>
<tr>
<td>Emil</td>
<td>Rasmus</td>
<td>M</td>
<td>A70257</td>
<td>N16</td>
</tr>
<tr>
<td>Jürgen</td>
<td>Mikk</td>
<td>M</td>
<td>B76449</td>
<td>N16</td>
</tr>
<tr>
<td>Andrei</td>
<td>Baranin</td>
<td>M</td>
<td>B42635</td>
<td>N16</td>
</tr>
<tr>
<td>Rudolf</td>
<td>Stenar</td>
<td>M</td>
<td>B30747</td>
<td>N10</td>
</tr>
<tr>
<td>Joosep</td>
<td>Soosaar</td>
<td>M</td>
<td>B80334</td>
<td>N10</td>
</tr>
<tr>
<td>Tarmo</td>
<td>Breedis</td>
<td>M</td>
<td>B75913</td>
<td>N10</td>
</tr>
<tr>
<td>Annika</td>
<td>Kuklane</td>
<td>F</td>
<td>B50979</td>
<td>N10</td>
</tr>
<tr>
<td>Carl</td>
<td>Peeter</td>
<td>M</td>
<td>B21279</td>
<td>N16</td>
</tr>
</tbody>
</table>

Keys column 3, values column 0:

<table>
<thead>
<tr>
<th>Enter key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>B75913</td>
<td>Joosep</td>
</tr>
<tr>
<td>B51296</td>
<td>Martina</td>
</tr>
</tbody>
</table>

Keys column 1, values column 4:

<table>
<thead>
<tr>
<th>Enter key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pilnik</td>
<td>N10</td>
</tr>
<tr>
<td>Marvet</td>
<td>N16</td>
</tr>
</tbody>
</table>
Solution

filename = input("Enter file name: ")
keyField = int(input("Enter keys column number: "))
valField = int(input("Enter values column number: "))

dict = {}
f = open(filename)
for row in f:
    parts = row.strip().split(',',)
    key = parts[keyField]
    val = parts[valField]
    dict[key] = val
f.close()

while True:
    print()
    userinput = input('Enter key: ')  
    if userinput == '':
        break
    if userinput in dict:
        print(dict[userinput])
    else:
        print("Not found")
## Dictionary application examples

<table>
<thead>
<tr>
<th>Objective</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dictionary lookup: find the definition of a word</td>
<td>word</td>
<td>definition</td>
</tr>
<tr>
<td>Book index: find pages where a given term occurs</td>
<td>term</td>
<td>list of page numbers</td>
</tr>
<tr>
<td>Web search: find web pages with relevant material</td>
<td>keyword</td>
<td>list of page names</td>
</tr>
<tr>
<td>File system: find file on disk</td>
<td>filename</td>
<td>location on disk</td>
</tr>
<tr>
<td>Compiler development: find properties of a variable</td>
<td>variable</td>
<td>type and value</td>
</tr>
<tr>
<td>DNS: find IP address of a domain</td>
<td>domain name</td>
<td>IP address</td>
</tr>
<tr>
<td>Routing table: route Internet packages</td>
<td>destination</td>
<td>best route</td>
</tr>
<tr>
<td>Financial accounting: process transactions</td>
<td>invoice No.</td>
<td>invoice details</td>
</tr>
<tr>
<td>Genomics: find markers</td>
<td>DNA string</td>
<td>known positions</td>
</tr>
<tr>
<td>Mathematics: perform operations with sparse vectors</td>
<td>index</td>
<td>element value</td>
</tr>
</tbody>
</table>
Exercises

https://courses.cs.ut.ee/2018/nkp/fall/Main/During11