AMS209 Homework 5 Documentation on Python

Xingchen Yu

Department of Applied Mathematics and Statistics,

University of California, Santa Cruz

November 2015

1 Introduction

In this homework, we incorporated dictionary object in Python to build a function that performs unit changes with respect to different metrics including meter, inch, feet, yard, nm, um, mm, cm and km. In the following sections, we describe our function implementation including inputs, outputs, usage of lists and dictionaries. In addition, we show several example outputs from implementing the function. This homework is edited in Latex using TeXworks as requested.

2 Description of Python Code

In this section, we illustrate the aforementioned Python code developed to convert a length in a given unit system into other systems and the corresponding Python code is shown as follows.
1. "Raw input" function is used to request user to input the length in given unit.

2. A dictionary is defined, of which each element is a pair of a unit and its corresponding value with respect to 1 meter.

3. The index number from the dictionary corresponds to input unit is found by looping through the keys in the dictionary. For example, if the input unit is "foot", then the index value for foot would be 1 since it’s the second key value in the dictionary.

4. All the values in the dictionary are normalized by the value in input unit which is immediately removed along with its key since the input unit is to be converted to other unit systems and not needed for the output.

5. The "unitchange" function multiply the normalized values in the dictionary with the input magnitude "length" and then each element in the output list is constructed in pairs with the output value and its corresponding unit.

Figure 1: Code Implementation.
3 Output Results

In this section, we show three outputs from implementing the code which are easily reproducible.

```python
>>> execfile("unitchange.py")
please input a length(number only)
> 100
Please type a unit system (meter, mille, inch, foot, yard)
> mille
'1.61030595813e+11 um', '528315.619968 foot', '161.83095813 km', '176185.1529
79 yard', '1.61030595813e+14 mm', '16103859.5813 cm', '6339787.27858 inch', '16
1830595.813 mm', '161030.595813 meter'
```

Figure 2: Code Implementation.

```python
>>> execfile("unitchange.py")
please input a length(number only)
> 10
Please type a unit system (meter, mille, inch, foot, yard)
> yard
['9144002.49448 um', '30.0000009144 foot', '0.00914400249448 km', '0.00367042554
987 mille', '9144002494.48 mm', '914.408249448 cm', '360.880108364 inch', '9144.0
08249448 mm', '9.14400249448 meter']
```

Figure 3: Code Implementation.

```python
>>> execfile("unitchange.py")
please input a length(number only)
> 3500
Please type a unit system (meter, mille, inch, foot, yard)
> foot
['1066799965.86 um', '1.06679996586 km', '6.662482778801 mille', '1166.66631197
yard', '1.06679996586e+12 mm', '106679.996586 cm', '41999.999332 inch', '10667
99.96586 mm', '1066.79996586 meter']
```

Figure 4: Code Implementation.