

## Assignment #17

## Sorting Data with Multiple Fields

Details of the names and ages of people may be stored in a list of tuples, such as:

```
[ ( "Ann", 23 ), ( "Tim", 18 ), ( "Bob", 37 ), ( "Ned", 51 ), ( "Sue", 18 ) ]
```

Write definitions for each of the following Python functions, and for each function, include a clear and concise comment to describe its purpose.

1. `Print( data )`

Output the list 'data' of (name, age) tuples in the form of a headed table, allowing 7 spaces for each name and 3 spaces for each age, with 1 extra space between them. For example, the output for the above list should be:

Name	Age
Ann	23
Tim	18
Bob	37
Ned	51
Sue	18

2. `InsertionSort( data, field )`

Sort the list 'data' of (name, age) tuples into ascending order of field 'field', which is either the string "name" or "age", using the Insertion Sort algorithm.

For example, calling `InsertionSort(data, "age")` on the above list, and then calling `Print(data)` should produce the output:

Name	Age
Sue	18
Tim	18
Ann	23
Bob	37
Ned	51

3. `SelectionSort( data, field )`

Identical to `InsertionSort(data, field)` but using the Selection Sort algorithm.

If multiple items have the same value for the sorting field, then sort the items in ascending order of the other field. Thus Sue comes before Tim in the example for `InsertionSort` above.

Program Submission:

Store the function definitions — *and nothing else* — in a file named 'a17.py', and turn it in for grading by typing:

```
submit-cs1117 a17.py
```

Due Date: Fri Feb 26, 10:30am