

## CSC 310 - Imperative Programming Languages, Spring 2010, Dr. Dale E. Parson

### Assignment 1, Mergesort in Python, due 11:59 PM, February 12, 2010

If you have not already copied my initial ProcLang/sortdemo subdirectory into your account area and verified that you have a working Python 2.6 environment as we went over in class, you must do that now.

```
python -V          # This should be version 2.6.2 on bill.
mkdir ~/ProcLang
cp ~/parson/ProcLang/sortdemo.zip ~/ProcLang/sortdemo.zip
cd ~/ProcLang
/bin/unzip sortdemo.zip
cd ./sortdemo
gmake clean test
```

This should work fine until you get to the mergesort test in the python subdirectory. That will fail because mergesort is not yet implemented.

```
python manysorts.py merge 1000 < ../manysorts.txt > manysorts.out
Traceback (most recent call last):
  File "manysorts.py", line 222, in <module>
    main()
  File "manysorts.py", line 57, in main
    sortfunc = mergesort
NameError: global name 'mergesort' is not defined
gmake[1]: *** [test] Error 1
```

Your assignment is to go into your copy of directory sortdemo/python, file manysorts.py, and add the code for function *mergesort*, which is currently commented out at the bottom of the file:

```
# STUDENTS: Implement mergesort as the Python mergesort function similar to
# the mergesort function in manysorts.cxx, with helper functions splitphase
# and mergephase defined inside of function mergesort (lexically nested).
```

```
# def mergesort(merger):
```

1. Implement this function using as pseudocode the C++ definitions of function *imergesort* and the helper functions that it calls in file sortdemo/manysorts.cxx.
2. You must combine C++ functions *imergesort* and *mergesort* into the single Python *mergesort* function. Python's *mergesort* function takes as a parameter a Python list to sort, similar to the other sorting functions in manysorts.py.

3. You can use a Python list as a queue (FIFO) in place of the FIFO classes in the C++ code. The FIFO operator to enqueue at the back of a list is the *append* function. The example list is this example is called *fifo*:

```
fifo.append(4)      # This enqueues the value 4 at the back of the fifo.  
  
fifo[0]           # This peeks the value at the front of the fifo.  
  
del fifo[0]       # This dequeues the value from the front of the FIFO.
```

Please note that “fifo” is just an example name for a variable bound to a list. It is not necessarily part of your program.

Also, where the C++ code constructs an array *splitter* of two empty `queueOfInt` objects in function *mergesort*, you can just do something like this:

```
splitter = [[], []]
```

*Splitter* is a list of 2 lists. You can use *splitter[0]* as one FIFO and *splitter[1]* as the other. *splitter[0][0]* is the front element of the first FIFO, and *splitter[1][0]* is the first element of the second FIFO.

Python’s *radixsort* uses a list of two FIFOs, referenced via variable *binarray*, in a manner similar to what you will need. Study it as an example.

3. Python helper functions *splitphase* and *mergephase* must be lexically nested inside of Python function *mergesort*. Lexical nesting is a way to keep helper functions private to the function such as *mergesort* that they help.

Python’s *quicksort* function has three lexically nested functions called *pickpivot*, *partition* and *helpquicksort*. Study how these are lexically nested inside of *quicksort* as an example similar to nesting of your helper functions inside of *mergesort*.

4. When “*gmake clean test*” works for all sorts including *mergesort* without diffs, then “*cd*” into your *sortdemo* directory and type this:

```
gmake turnitin
```

If it completes without an error message (after it prompts for a Carriage Return that you must enter), then your assignment has reached me. If you later make changes and want to turn in a revised assignment, just “*gmake turnitin*” again from the same directory. It will over-write any previous submission with the new one.

Please start early, so you can come to see me with the problem if you get stuck.