SEng 265: Programming Exercise 2

Problem Summary

Your task is to produce a new Code Activator quiz based on three sorting algorithms: bubble sort, insertion sort, and selection sort. Your quiz must contain 18 questions, randomly selected from a pool of 45 questions.

This is a team assignment; it must be completed by the teams formed in lab.

Downloading the files

1. Use your browser to download the C and Code Activator source files.
2. From the directory where you have placed the zip file, unpack the files and generate the code library by typing:

```
unzip exercise2_download_v1.1.zip
cd exercise2_download_v1.1/codeAct_1.01
cd generate
python ../bin/generate.py min/min_io.py ../code_library
python ../bin/generate.py min/min_bull.py ../code_library
python ../bin/generate.py min/min_liar_liar.py ../code_library
python ../bin/generate.py min/min_ff.py ../code_library
```

3. Test the installation by typing

```
cd ../web2py
./start.sh &
```

   and opening your browser with URL:

```
http://localhost:8080/codeAct/static/ca_index.html
```

   Then click on the link for the Input/output question, answer the question correctly, click Submit, and confirm that Code Activator indicates "correct".
4. Study the code for the C source code files `exercise2_download/`:

```
bubble_sort.c
insertion_sort.c
selection_sort.c
```

Develop the question templates

There must be three question templates for each of the three sorting algorithms:

Count the number of swaps
: This template asks the student to predict the number of swaps performed

for an invocation of the sorting algorithm.

Follow the smallest element
: This template asks the student to predict the array index, after each swap, of the smallest array element.

Find the failure
: This template asks the student to produce a inputs which will reveal a non-trivial fault in the sorting algorithm.

There must be five questions generated from each of the nine question templates, for a total of 45 questions in the code library. Each template must use one of the three sorting functions in the downloaded files with as few changes as possible.

The directories containing the generated questions in the code library must be named T_N, where T is the name of the question template file, e.g., `bubble_sort_swap_count` and N is in [0..4].

Develop the question specification

The quiz specification must run in practice mode. From each of the nine question templates, there must be two questions randomly selected, for a total of 18 questions each time the quiz is run.

Grading Criteria

Your mark will be based on the following criteria:

[50%] Required question templates and quiz specification present
- Have you provided the required question templates and quiz specification?
- Do the question templates focus on the required topic?

[14%] Solvable
- Does every question have at least one correct answer?
- Could a typical 265 student achieve a mark of at least 80% in 30 minutes, after one 30 minute practice session?

[14%] Crash-free
- Is each question guaranteed to terminate without addressing errors, regardless of student input?

[14%] As simple as possible
- Are your question templates as simple as possible?
- Are all of the question templates less than 100 lines in length?

[8%] Assignment submission
- Have you followed the instructions for submitting the assignment?
- Were you able to successfully submit with the first attempt?

Submitting your Solution

You must submit a single zip file

```
N.quiz.zip
```

where `N` is the logins, '_' separated, of the team members. Be sure that these are the logins used in the SEng 265 lab.

The zip file must contain the following files and no others:

```
sort_quiz_spec.py

bubble_sort_swap_count.py
bubble_sort_min_position.py
bubble_sort_bug.py

insertion_sort_swap_count.py
insertion_sort_min_position.py
insertion_sort_bug.py

selection_sort_swap_count.py
selection_sort_min_position.py
selection_sort_bug.py
```