Assignment 2

Experiment for 2.3

See *experiment_2_3.cpp* to view the code for this experiment. We made a record of 100 strings, each of size 10, which looks like this:

Record record =
{"AAAAAAAAAA",
"AAAAAAAAAA",
"AAAAAAAAAA",
"AAAAAAAAAA",
… 96 more times…

Since we have 100 attributes and each being 10 bytes each, the record above should take 1000 bytes to serialize.

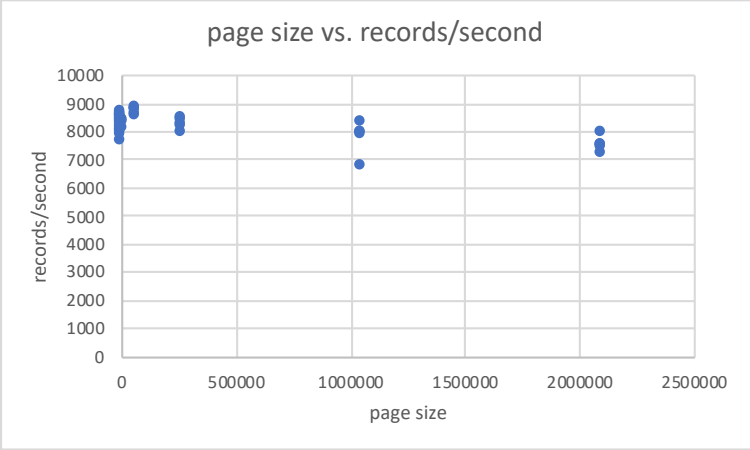
Surely, after running *fixed_len_sizeof()*, we got 1000 as the answer ☺

```
b2210-14:~/Desktop/DataLayout$ experiment_2_3
1000
```

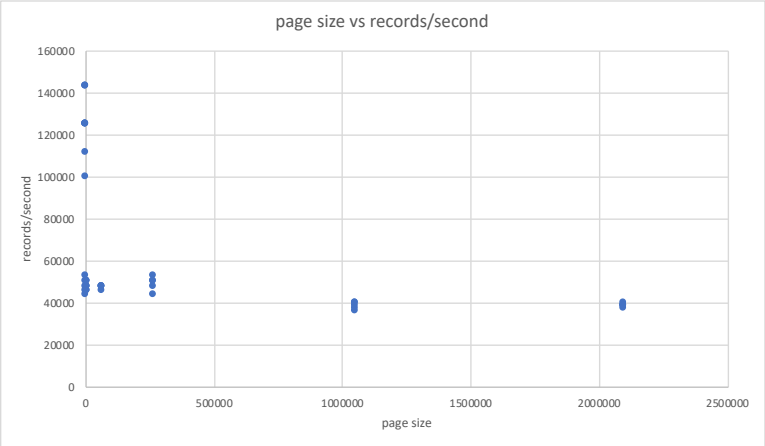
Experiment for 3.2

See *experiment_3_2_write.py* and *experiment_3_2_read.py* to view the code for this experiment. You can view the page sizes used in those files.

Here is the plot for *write_fixed_len_pages*



Here is the plot for *read_fixed_len_page*



As we can see, storing records in a page-based format has way better performance. This is because of many things:

- Page files can store values in binary format whereas csv files can only store values in strings
- Deleting records in csv files is very slow, rows must be shifted when a row is deleted, whereas page files can easily delete a record by changing the page's directory entry
- Page files don't need to store commas and new lines, thus saving space compared to csv files

One shortcomings of the way we organize pages is that records for us must have a fixed size, or else they can't be addressed using the bitmap that we use.

Couldn't finish the following

Experiments for 4.3

Experiments for 5.2