1. Introduction: A section that explains the essential problem of page replacement and briefly summarizes the structure and implementation of your simulator. Do not copy and paste text from this project description. In your own words, describe the overall structure and purpose of the experiment.

2. Methods: A description of the experiments that you performed in order to learn something about each memory trace. Of course, it is impossible to run your simulator with all possible inputs, so you must think carefully about what measurements you need to answer the questions above. Make sure to run your simulator with an excess of memory, a shortage of

  4

memory, and memory sizes close to what each process actually needs. For instance, you may want to think strategically of what values you use for the <nframes> parameter. On one hand, you want to cover enough of the memory space to see when (a) your "process" does not have enough physical memory(thus, a lot of misses!), (b) when your process' working set fits in the memory, and (c) where increasing the allocated memory does not improve performance significantly. On the other hand, the more values for <nframes> you test with, the more hours you'll spend starring at the computer. To help you decide, think of number of frames as power of 2. So perhaps a sequence of 1, 2, 22, 23, 24, ..., 210 might be better than 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, for example. Also, think of what <nframes>x<frame size> means for today's processes -- perhaps an allocation of 2 TB of memory for a process is not particularly common these days (and neither is a physical memory of 32KB).

3. Results: A description of the results obtained by running your experiments. Present the results using both tables and graphs that show the performance of each algorithm over a range of available memory. For instance, include a graph of hit rate vs. cache size for each algorithm. In the text, summarize what each graph or table shows and point out any interesting or unusual data points. Feel free to focus your report on one trace only. For that trace, give results and discuss in detail the performance of the 4 algorithms. Doing an outstanding job on simulating the addressing of memory on one trace only will give you full credit. If you have more time, identifying and comparing the sizes of the working sets of multiple traces might give you another point to discuss in the report.

4. Conclusions: Describe what you have learned from the results. What have you learned about the memory traces? What have you learned about the paging algorithms? How does the size of available memory affect memory performance? Be sure to describe clearly how specific results above lead you to these conclusions.