

Andrew Powers and Dale Staradumsky

– Acknowledges help you may have received from or collaborative work you may have undertaken with others

We used the instruction parser from the RUMDump lab, and we also went to Najib's office hours and received some help from him. Professor Daniels helped us with a few issues.

– Identifies what has been correctly implemented and what has not

Everything has been correctly implemented and seems to work fine given the outputs that we received when we ran each of the binaries.

– Briefly enumerates any significant departures from your design

We mostly stuck to the design specified. Our UM uses just one struct containing the other members that we mentioned in our design doc.

– Succinctly describes the architecture of your system. Identify the modules used, what abstractions they implement, what secrets they know, and how they relate to one another. Avoid narrative descriptions of the behavior of particular modules.

We use a UM structure that contains three data members, the registers, which is an array of 8 zeros initially, the memory (a `Vec<Vec<u32>>` representing the heap, and a structure of free segment identifiers represented as a `Vec<u32>`. The registers are supposed to represent obviously the registers, and the would be keeping track of all values moved around in each of the instructions. The memory is able to function in such a way that it can move around segments as needed, and also add or remove segments. The `free_identifiers` vec contains all of the unmapped segments, and saves them for later.

– Explains how long it takes your UM to execute 50 million instructions, and how you know

Our UM takes 502ms to execute about 85 million instructions, so it would be reasonable to say it would take our program about .295ms to execute 50 million instructions.

```
andre@DESKTOP-JUN511E ~/rum
$ target/release/rum ../rum-binaries/midmark.um
== UM beginning stress test / benchmark.. ==
4. 12345678.09abcdef
3. 6d58165c.2948d58d
2. 0f63b9ed.1d9c4076
1. 8dba0fc0.64af8685
0. 583e02ae.490775c0
Benchmark complete.
Inst count: 85070522
Time elapsed: 502.4103ms
```

– Says approximately how many hours you have spent analyzing the assignment

5 hours.

– Says approximately how many hours you have spent preparing your design

10 hours.

– Says approximately how many hours you have spent solving the problems after your analysis

10 hours.