

Programming Assignment 1

CSCE 313-503

2/9/2018

Khanh Nguyen

UIN# 525000335

Prof. Tanzir Ahmed

The program uses the implementation of the ackerman function to test the memory allocation. The running varies for different for each set of value n and m.

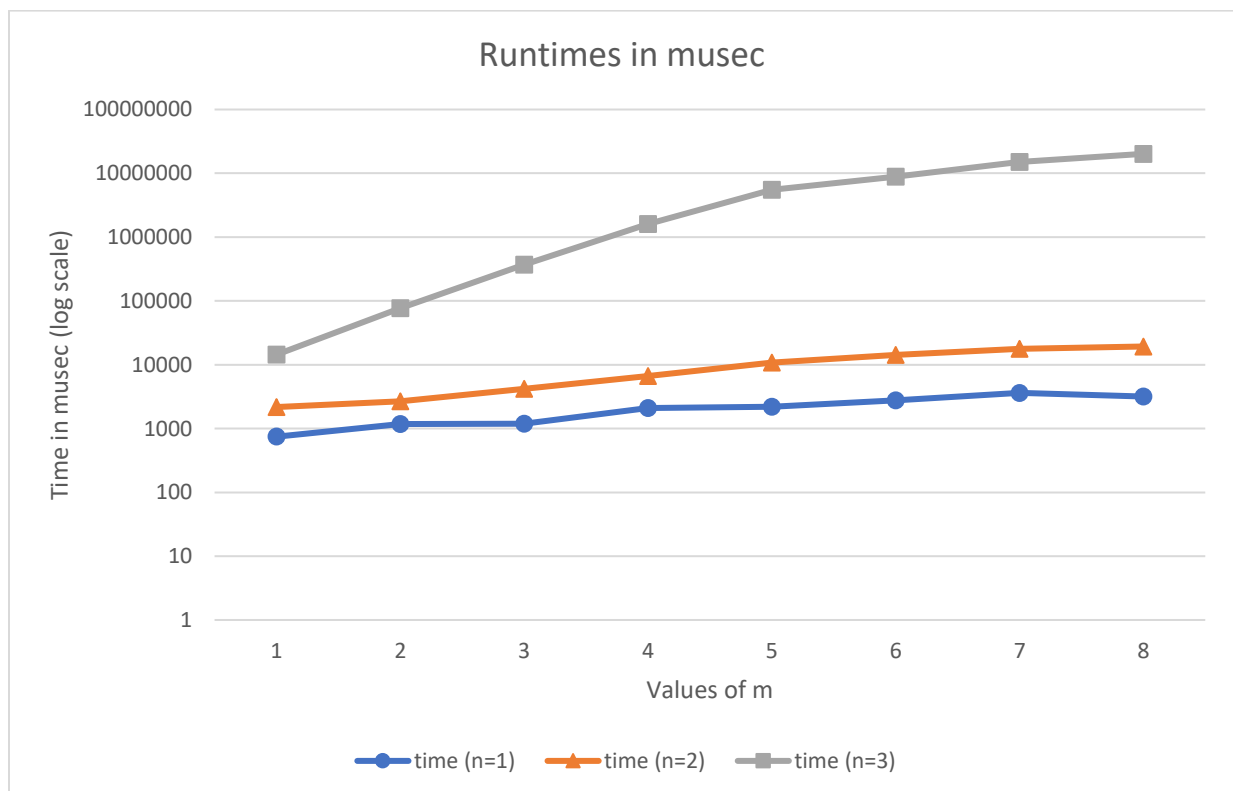
Result tables:

m	time (n=1)
1	746
2	1180
3	1188
4	2091
5	2187
6	2778
7	3626
8	3178

m	time (n=2)
1	2173
2	2659
3	4189
4	6615
5	10809
6	14265
7	17677
8	19313

m	time (n=3)
1	14373
2	76805
3	371277
4	1598149
5	5495698
6	8892071
7	15147668
8	20171238

The graph below is the result of running time (y-axis, log scale)in micro seconds for each fixed value of n with m (x-axis) from 1 to 8:



It can be easily observed that there's not much different in term of time when $n=1$ or $n=2$. However, when $n=3$, the running time increases significantly as m increases.

The bottleneck in the system can be considered that when a large amount of memory is initialized, and small memory is requested, it would take a lot of time to split the large free blocks into smaller pieces. Also, because the header files have the same amount of memory for each block, if there are too many small blocks in the list, those header files will take a lot of space in memory. Therefore, it's only effective when the memory requested is large.

One possible improvement for this system is that it should only join the block when there is not enough memory to allocate in that block. That change would reduce the cost of joining blocks for a system that runs many small programs that only need small blocks. However, it's a trade-off since the implementation of finding buddies will be harder.

Instructions to run the program:

make all

./memtest