# 15869 Homework Assignment 2:
# Due at the start of class Thursday, Nov 22

Being able to distill complex systems and algorithms into simple explanations of their core features and characteristics is a useful skill in all of computer science, but particularly in system design. This assignment will test your ability to do so. Also, as an architect, you should be willing and able to understand the basic computational structure of complex applications (even if the details or science behind the applications is unfamiliar to you). Thus, this assignment involves domains not previously discussed in class. However you can be assured that the same design principles we've discussed in the context of graphics still applies.

In this assignment you are to read one of the three following papers and describe the paper in a concise, one-page write up. You should not think of this as a paper summary. Your goal in the write-up should be to teach another member of the class about the key ideas of the system in less than five minutes. In fact, you may have to! On Tuesday I will be randomly picking from the class to perform such five-minute presentations. Providing a figure to assist your description is likely a wise choice (hand-drawn is just fine) but your entire write-up still has to fit on one, single-sided page.

Paper 1:

D. E. Shaw et al., Anton: a Special-Purpose Machine for Molecular Dynamics Simulation. ISCA 2007

D. E. Research has designed a supercomputer, largely from the ground up, with very specialized hardware to run molecular dynamics simulations. Explain the basic architecture to us.

Hints:
What is the problem this machine is trying to accelerate? (note comments about number of time steps)
What is the primary challenge with this workload?
What is the "High-Throughput Interaction System" for?
How are processing elements connected? What entities are communicated?
Is bandwidth or latency a bigger concern for this workload?
Are there any parallels with optimized implementations of the graphics pipeline?

Paper 2:

D. E. Shaw, A Fast, Scalable Method for the Parallel Evaluation of Distance-Limited Pairwise Particle Interactions. Journal of Computational Chemistry. 2005.

This algorithm for computing particle-particle interactions was developed with the Anton machine (see above) in mind. Explain the algorithm and its major benefit.

Hints:
What is the problem this algorithm is trying to solve? Where is the communication in this algorithm?
What is the unintuitive idea at the heart of this approach?
Recall how we've had considerable discussion of compute-to-bandwidth ratios in class?
This paper is about asymptotics. Your explanation should be build around them.

Paper 3:

Z. DeVito et al., Liszt: A Domain Specific Language for Building Portable Mesh-based PDF Solvers. Supercomputing 2011.

In class we've talked at length about the relationship to graphics pipeline abstractions and their implication to the programmer and the system implementer. Liszt is a new domain-specific language that seeks to simplify development of codes for solving PDEs on meshes. It's largely intended for supercomputing scales. Explain the Lizst programming model.

Hints:
What are the primary abstractions?
What are the entities?
What are the key optimizations that these abstractions enable?
You may wish to peruse the Lizst web site (or read the SCIDAC paper for more perspective)