

Santa Clara University  
ELEN513  
Final Exam

Dr. Hossein Omidian

June 2023

## 1 EXAM:

This exam contributes towards your final grade. You are to submit it electronically using Camnio. Please make sure that you use a word processor to produce your submission. For example something like MS Word or Latex. Submit either a pdf or the MS Word file as well as the archive of your project and all the python code or extra code you used for this exam. Please include a README file and describe steps needed to run your code.

1. **[20 marks] Parser:** You need to write a parser code. Your parser will parse a simple code and generate a simple Data-Flow-Graph as well as Intermediate Representation output. DFG should be dumped as text and also as an image (using Graphviz is recommended).
2. **[35 marks] Multi-core backend:** You need to write a backend code which generates backend codes for different size multi-core systems. To make things simple, we can assume the multi-core system is a simple system with multiple PEs and fully connected interconnect or shared memory so no need to take care of placement or optimize based on the location of PEs. Your backend gets the IR and number of PEs and generates an optimized and balanced implementation and dumps output code for each PE. You can use a greedy approach such as using Integer Linear Programming or a heuristic approach. Operation latency should be read in as a separate text file (e.g., JSON or XML).
3. **[15 marks] QA** You need to write a simple simulator/performance analyzer type code to make sure all the files generated by your backend work together and generate the same output as the scalar code.

What to submit:

1. A document file describing your algorithms and code and how you solve each problem.

2. An archive of all the source codes, run files, makefile and, ...
3. README file

## 2 Example

Listing 1: Simple code.

```
1  t1=LOAD(x);  
2  t2=t1+4;  
3  t3=t1*8;  
4  t4=t1-4;  
5  t5=t1/2;  
6  t6=t2*t3;  
7  t7=t4-t5;  
8  t8=t6*t7;  
9  STORE(y,t8);
```

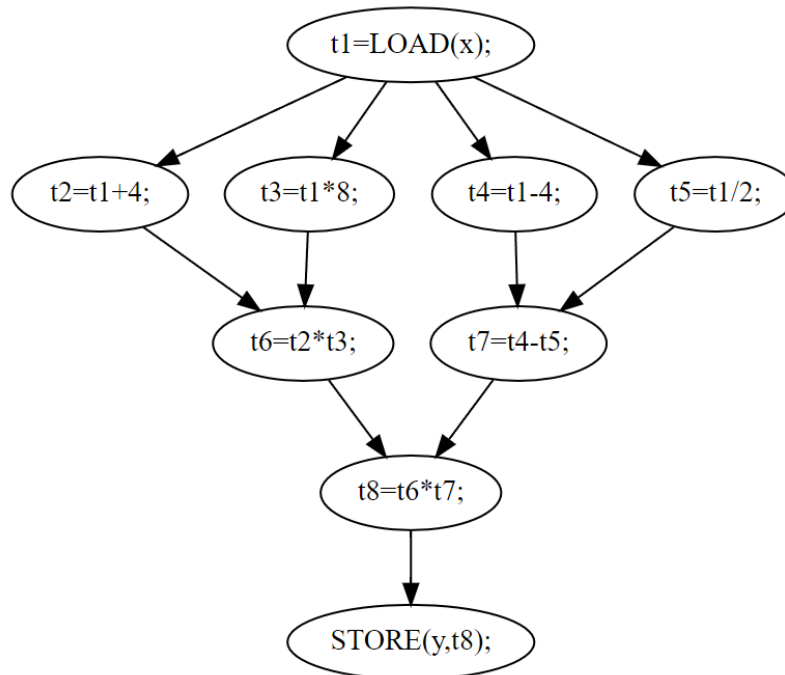


Figure 1: DFG