CS 366  
Assignment #7  
Poly want a nomial ... fast!  

Due 4/25/18, in class

Goal  
Study processor-level optimization.

Problem Statement  
Our lads, Sponge Bob and Patrick, have a polynomial system to beat the stock market as long as they can evaluate polynomials fast! A polynomial such as \( a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n \) can be evaluated given coefficient values, \( a_i \), and a value for \( x \). There is a need for speed.

Your task is to create a faster version of the code found in the repo using the optimization techniques explored in Chapter 5 (e.g., loop unrolling, parallel accumulation, re-associations, etc.).

What to hand in  (Please no .docx files.)

1. A well-formatted 2-up postscript file of your source code checked into your repository named `src.ps`. You must use `a2ps` after removing all the tabs from your code. Indent code 2 or 4 spaces at most.

2. A GitHub repo that includes (you must use these names as the grading script will assume their use!)
   - `README.md`,
   - `poly.c`, and
   - `Makefile` (where `make all` and `make test` will build and test your code).

Assignment Notes

- Please make changes one at a time. Document each change in `README.md` with five things
  0) a section title with a good name,
  1) a description of the change,
  2) the motivation for the change,
  3) a copy of the function `poly()` that results from the change, and
  4) the CPE output line from the run (copy and paste this).
- FYI, the test harness generates random values for each \( a_i \), using \((\text{random()} \& 0x1) ? 1 : -1\) and uses the value 3 for \( x \).
- I expect to pull your code, run `make`, and then run my test script.
- The output of `git log` will again factor into your grade.
- The book’s author’s note “Ideally you should be able to reach a CPE close to the throughput limit of your machine. Our best version achieves a CPE of 1.07 on our reference machine.”
- Here is the git classroom invitation  
  https://classroom.github.com/a/vLUbU6q
- Five, count-em 5 bonus points to the program that achieves the lowest CPE!!!
- Check out Practice Problems 5.5 and 5.6!