

C  mpiler

C  nstruction



Simone Campanoni  
simone.campanoni@northwestern.edu

# Competition



# Competition: how to win

- The team that has designed and developed the LBC that generates the fastest binary for the target LB program wins the competition!
- Target LB program: competition2023.b  
I will release it this coming Sunday

# Competition

- It will be live
- All correct LB compilers will participate together with gcc, clang, and my compilers
- Use my compilers to understand whether you have optimized the generated code enough
  - Every year so far:  
the winner generates a faster binary compared to the one my compilers generate

# Workload for the class competition

- LB program: competition2023.b
- After compiling this program, take a look at L1/prog.S 😊
  - Could you have implemented the same workload writing it directly in assembly ?
- The official competition will use only your compilers that you have submitted **before** the competition deadline
- Deadlines:
  - See Canvas

# Competition: how to join the competition

You can compete only if

- Your compilers pass all L1, L2, L3, IR, LA, LB tests
- You submitted all tests for all languages and they are all correct in the latest framework (make sure to include the .in files for ALL of your tests -- resubmit if necessary)
- You submit your LBc before the deadline (hard deadline)
- Your LBc compiles competition2023.b in less than 10 minutes on hanlon

**Good luck!**

# What todo with your LBc

- Testing:
  - Compile all compilers:  
run `make -j` from the parent directory of the framework
  - Test all compilers  
run `make test` from the parent directory of the framework
- Submitting
  - Please use “make homework” to upload your work
- Competition: the compiler that generates the fastest binary wins
  - Login in hanlon
  - Compile your compilers: `make` from the parent directory
  - Run your compiler: `cd LB; make performance`

Always have faith in your ability

Success will come your way eventually

**Best of luck!**