PA4 – Memory System: Part A & Part B

Due Date

- See Piazza for due date and time
  - Grading the next day
- Submit program to perforce in your student directory
  - Sub directory called:
    - /PA4/...
  - Fill out your ***PA4 Submission Report.pdf***
    - Place it in the same directory as your solution
    - Enter the final Changelist number of your submission
    - Enter the number of test passed
      - Requires 15 tests to be passed from the supplied tests
      - Stress test
    - Write up a quick discussion in the report
      - What you learned from this assignment so far

Goals

- Learn
  - To Create a Memory System from scratch
- Understand the internals of a memory system

Assignments

1. ***Create a memory system within a heap***
   - Take the given memory system framework for the heap layout:
     - Add the allocators
     - Add the de-allocators
   - Run the Test functions that handles a set of memory allocation and de-allocations
     - *Supplied by Instructor*
     - Part A & B:
       - tests 1-15 completed
       - stress test
     - Copy your work from PA3 and move it to PA4
   - Diagram the data structure layout out - to help you.

2. ***Take Memory system, use the stress test***
   - Measure the timing with default setting in the compiler
     - For the original memory system
     - For your custom memory system
   - Measure the difference.
     - Write the before and after in an output text file
   - Instructor will provide the stress tests
     - Make sure your program runs all unit tests 1-15
     - Make sure it runs the stress test

General:

- Write all programs in cross-platform C or C++.
  - Optimize for execution speed and robustness.

- Create a programming file for each problem, for example
  - Student directory
    - /PA4/memory/…
  - Make sure that each problem can be compiled and run through the checked in solution

- Do all your work by yourself
  - Feel free to talk with others about ideas and problems
  - But do not copy your friend's code.
    - Please don't - I can tell with my difference tools
  - Feel free to share ideas

- Check in the problems multiple times
  - At least 3 times per problem
    - For this project it might be closer to 40
  - Have reasonable check-in comments
  - Seriously, I'm checking

- Make sure that your program compiles and runs
  - Warning level 4, some times that is not possible due to MS headers…
  - Your code should be squeaky clean.

- We are using Perforce
  - You should have received the document describing how to login.
    - Please look at the documentation and videos under the reference directory
  - Submit program to perforce in your student directory
    - Sub directory called: /PA4/…
      - As described above
  - All your code must compile WARNING FREE from perforce with no modifications.
    - Otherwise it's a 0, no exceptions
  - Only Visual Studio 2013 allowed

Validation

Simple check list to make sure that everything is checked in correctly

- Do they compile and run without any errors?
- Submitted it into /PA4 directory?
- Did you do the submission report?
- Can you delete you local drive, regrab the PA4 directory?
    - Is all the code there?
    - Does it compile?

Hints

Most assignments will have hints in a section like this.

- Do many little check-ins
    - Iteration is easy and it helps.
    - Perforce is good at it.
- Look at the lecture notes!
    - A lot of good ideas in there.
    - The code in the examples work.
- Use the Piazza
    - This is much harder than the last assignment.
    - See me during office hours.
    - Read, explore, ask questions in class