

# CAAM 420 Fall 2011 Homework 3

T. Warburton

September 16, 2011

You must complete the following task by 5pm on Monday 09/19/11.

Your solutions to the homework must be committed to your Subversion repository in a sub-directory `HW03`. You will commit your files as follows:

- i. a text write up formatted as a pdf file in `HW03/hw03.pdf`
- ii. your source code files in `HW03/main.c`, `HW03/mysin.c`, `HW03/mycos.c`, `HW03/myasin.c`, `HW03/header.h`
- iii. your Makefile in `HW03/Makefile`

**Q1 Purpose:** practice programming, testing, debugging math functions and using prototypes, header files and the `gnu make` system

In homework 02 you used partial sum approximations to numerically approximate the sine function

$$s_N(x) = \sum_{n=0}^{n=N} \frac{(-1)^n x^{2n+1}}{(2n+1)!}.$$

In this homework you will use Newton's method to evaluate an approximate inverse of the sine function, commonly referred to as the arc-sine function. As the number of functions in a code project grows there is an evident need to manage the source files. To this end you will create a separate source file for each function. To extend the scope of each function you will create a header file that contains a prototype for each function (with the exception of `main`).

Homework tasks:

- i. [10 points] Create a header file `HW03/header.h` containing prototypes for your `mysin`, `mycos` and `myasin` functions.
- ii. [10 points] Create a C-file called `HW03/mysin.c` containing your implementation of partial approximation to the sine function from homework 02.
- iii. [10 points] Create a C-file called `HW03/mycos.c` that implements the cosine function in a similar way to how you implemented `mysin`.
- iv. [10 points] Create a C-file called `HW03/myasin.c`. In this file you will implement a Newton iteration based method for finding an approximation to the arc-sine function. Namely given  $y$  find an  $x$  such that  $|\text{mysin}(x) - y| < \text{tol}$  for some user supplied tolerance.

**Note:**

- a. You should use your `mysin` function and optionally your `mycos` function in the evaluation of the Newton step. Do not use the math library "`sin`" function.
  - b. You are computing the approximate inverse of a function that computes an approximation to the `sin` function. i.e. there are two approximation errors being accumulated.
- v. [10 points] Download the common `main.c` file here [main.c](#) . Save `main.c` in your `HW03` directory:
  - vi. [10 points] Download the Makefile from [Makefile](#). Save it in your `HW03` directory and make sure that it correctly compiles your code with the `make` command.
  - vii. [10 points] Make and debug your code. Document your results as a pdf formatted paper stored in `HW03/hw03.pdf`. Explain any deviation of the output from your `main` function from expected behavior.
  - viii. [10 points] Describe briefly the debugging process you used to make sure your functions behave as specified. In particular describe any use of `gdb`.
  - ix. [10 points] Add and commit your files to a subdirectory `HW03` directory in your CAAM420 Subversion repository.
-