Crash Course in C

Programming Exercises

Exercise 1

- Take a number as a command line input from the user and print it
  - You will find the atoi function in stdlib.h useful. It converts a character array to an integer
  - Make sure you check if the user passed the argument (check the value of argc)

- Compilation instructions
  - gcc myprogram.c –o myexec
  - Also try out gcc –S myprogram.c (will generate myprogram.s)

Exercise 2

- Declare a static array of size 10 and generate the first 10 Fibonacci numbers
  - A[0] = 0 and A[1] = 1.
  - Thereafter A[i] = A[i-1] + A[i-2]
  - Your output should be 0,1,1,2,3,5,8,13,21,34

Pointers

- int x = 10, y = 25; // declare two ints
- int *p = NULL, *q = NULL; // and two pointers to ints
- p = &x; // p now points to x
- printf("%i\n", p);
- printf("%i\n", *p);
- p = &y; // p now points to y
- printf("%i\n", *p);
- q = p; // q now points to the same thing p does
- printf("%i\n", *q);
- *q = 9;
- printf("%i\n", *p);

Exercise 3

- Change the program you wrote for the previous exercise to do the following
  - Size of the array = n is a command line input
  - Dynamically allocate memory for the array
  - Compute the first n Fibonacci numbers
  - Make sure you free the memory you allocated
  - If you are up for a challenge, use pointer arithmetic to navigate the array instead of indices