National University of Singapore
School of Computing
CS1010X: Programming Methodology
Semester II, 2019/2020

Mission 17
More C Madness

Release date: 09 May 2020
Due: 07 June 2020, 23:59

Required Files

- mission17.zip

Information

This mission will give you a taste of how programs are written in C and hopefully also an appreciation of the differences between C and higher-level languages like Python. There are 2 distinct tasks, each with one or more tasks to complete. When you are done, copy only the function(s)/struct(s) you were asked to modify or implement into Coursemology.

Getting Started

Programs in C can span across multiple files, which will need to be compiled as a whole. Task 2 is one such a program. To compile it, run:

```
$ gcc -Wall -o psort psort.c data.c
```

Task 1: Spiraling Inwards (5 marks)

Files: `spiral.c`
Items: 1

Given a square matrix (2D character array) of specified length and a string of text, fill up the matrix with the text, starting from the *top-left* corner, in a *clockwise* direction, from the *outside in*. The text should be repeated over and over until the entire matrix is filled. You may assume that the length of the matrix is always a positive even number.

Task 2: People Processing (5 marks)

Files: `psort.c`, `data.c`, `data.h`, `people.txt`
Items: 2

This task spans over 3 files. The `data.*` files define a module for loading information from a data file (`people.txt`) into an array of pointers to Person structures. The `psort.c` file contains the main function and imports the data module for usage.

Your first subtask is to add a field for storing the Body Mass Index (BMI) of a person. You will need to complete the structure in `data.h`, the print function in `data.c` and finally a function for computing and assigning the BMI for every Person in the array in `psort.c`.

The second subtask requires you to implement any one of the sorting algorithms you have learned to sort the data array. The `sort_data` function must support sorting over any of the fields in the Person structure, and in either ascending or descending order. You are NOT allowed to use the `qsort` function provided by C.