Tushar Mohan
2019393

Purpose:

This program is intended to read a csv file containing data of numerous students containing their student ids, sections and marks of their 4 assignments, and then compute the average marks of each student and display the computed results. The computation for each student belonging to section A is done by a child process which is created via fork() systemcall while the computation for the section B students id done by parent process itself.

In-built functions and system calls used :

- **fork()**
  - It is used to make a new process called child process.This child process concurrently with the process which calls the fork() (namely the parent process).
  - It takes no parameter.
  - Return an integer which indicates the whether the process creation was successful or not.
    - Negative value means that fork() call was unsuccessful
    - Zero means that the child process was created
    - Positive value is returned to the parent or caller. The value is nothing but the ID of the newly created process.

- **waitpid()**
  - It takes 3 parameters.
    - First is the pid which is returned by the fork().
    - Second parameter can be NULL as in this case
    - Third parameter can be zero as in this case.

  - It is used to make the parent process wait while the child process executes and then the parent process is resumed.

- **exit()**
  - Generally a status code is passed into it to show success or failure of your program.
  - In this case, it is 0 which indicates success.

- **fopen()**
  - It is an in-built function that is used to open the csv file.
  - It takes 2 parameters
    - First is the file path
    - Second is the mode you wish to open in which can be “r” for read, “w” for write and “a” for append.
  - It returns a file pointer which can be used to perform necessary actions.

- **fclose()**
    - It is used to close the file pointer.

References:

- https://www.geeksforgeeks.org/fork-system-call/