## Programming Exercise D1 - Two Children

### Assignment Objectives

* Practice using the fork() function to create two child processes

### Assignment Summary

In this assignment, you will finish a partially-written C program that uses two child processes to print the contents of an integer array in forward order (done by one child process) and in reverse order (done by the other child process).

The **sample-runs.txt** file contains a series of sample runs of the program. The output from your program should match the format and style of the child process line shown in these sample runs. The rest of the lines are printed by the parent process.

### Directions

1. Download the Two-children-student.c file and change the name to Two-children.c
2. Read through the source code. The main() and parseCommandLine() functions are already written, so make no changes to these functions. The rest of the function definitions are stubbed out
3. Implement the following functions:
   * **void fillTable(int list[], int nbrOfEntries)**

This function fills the list with random integers in the range from 0 to 99

* + **void generateChildrenProcesses(int list[], int nbrOfEntries);**

This function contains a for loop that calls the fork() function to create the two child processes. One child process should call the displayTableForward() function and the other child process should call the displayTableReverse() function

* + **void displayTableForward(pid\_t processID, int list[], int nbrOfEntries);**

This function displays the integers in the list in forward order

* + **void displayTableReverse(pid\_t processID, int list[], int nbrOfEntries);**

This function displays the integers in the list in reverse order

1. Submit your completed source code file on the Blackboard course website