IC221 Lab: Memory Leaks Worksheet Name FRANCESCONI

Sping AY2022, 100 points total

**Task 1 (50 points)**

(5) Compile and execute memleak.c. Verify the output and review the program.

(10) Run valgrind on the memleak program. How many bytes does it say have been “definitely” lost?

|  |
| --- |
| 60 bytes |

(5) On what line(s) of code does valgrind indicate a memory leak has occurred? 34, 50

(10) Identify and describe at least one memory leak in memleak.c.

|  |
| --- |
| on line 34 memory was assigned to pointer a but it is not freed before the end of the program  on line 50 pointer a to the original array is lost when it is overwritten with a pointer returned by doubleup(). |

(10) Fix the memory leak you identified and verify your fix with valgrind.

(10) Describe how you fixed the memory leak:

|  |
| --- |
| Inside method double up I freed pointer a at the end of the function before the new pointer is returned to overwrite a in main. I also freed a in main before the main method ends. |

**Task 2 (50 points)**

(5) \_\_\_\_\_ Compile and execute the memviolation.c program.

(10) Describe the output and exeuction of the program. Does it seem to be consistent?

|  |
| --- |
| Yes, it always prints out Hello World! |

(10) Run the program under valgrind. Identify the line of code that is causing the memory violation and its input:

|  |
| --- |
| printf("%s\n", str); |

(15) Describe the programming bug:

|  |
| --- |
| The string str isn’t terminated by a null character |

(10) \_\_\_\_\_ Fix the memory violation and verify your fix with valgrind.

**Submission**

- Fixed memleak.c

- Fixed memviolation.c

- This completed worksheet