

Exercise 4: Memory Management, File Systems

Task 1: Memory Management

T1-1: The array can't be initialized on the stack, it is too big, so the easiest solution is to allocate it on the heap.

T1-2: The update function tried to access indices of array to that were not in range of SIZE (the random generated numbers in *init()*). To solve the problem I added the modulo operator, so that the index is valid. I don't know a condition to update the value at a specific index, but this change, solves the problem without changing anything outside the update function. . .

T1-3: *CLASSIC MEMORY LEAK*: The problem was that the *count_sort* function calls *calloc()* for *temp*, but never frees the memory again. If you call it in loop, the program allocates every time memory for a new *temp*, but does not free the memory of the old *temps*.

Task 3: Virtual vs. Physical Memory

Virtual memory is the memory that a program needs and its then mapped by the OS to a part in the physical memory. Typically this physical addresses are saved in page tables, where the MMU looks and adding the offset of the page table, calculates the physical memory address for the program.