Noel Ball
CSE13S

sorting.c Design Document

Description:

An executable test harness sorting takes command line arguments to print different sort functions, potentially with specific seeds or sizes, statistics including the number of elements in the array, and a help menu. The sort functions are included in the files; batcher.c, shell.c, heap.c, and quick.c.

Goal:

Run ./sorting with chosen arguments, and getopt() parses the input argument. A switch-case statement is used to update a set depending on which sort functions need to be run and if statistics are included, as well as track inputs such as specific seeds or sizes. srand(seed) is used to set the seed, and rand() is used to create an array of random bit-masked numbers during a for loop. Then a set of if-statements run the functions to sort the array and format the data to print.

Sudo:

Batcher sort in batcher.c uses a comparator function that swaps two elements if the first is greater. The sort starts by counting the bit length of the values in the array, followed by the provided pseudocode from the documentation.

Shell sort in shell.c uses the gaps header file to calculate the length of the sort at a given time. The sort starts by looping through all the gaps ranges, excluding the ranges larger than the values

in the array. It then iteratively compares elements based on the pseudocode, performing swaps and reducing the gap until the final array is sorted.

Heap sort in heap.c uses functions to build and maintain a max heap including max_child, fix_heap, and build_heap based on the pseudocode. The sort then builds the max heap, to move the largest data to the bottom to be removed from the heap. The final array when the heap is empty is sorted.

Quick sort in quick.c uses a partition function to create a pivot to compare values against, sorting the array into two subarrays; one less than the pivot and the other greater than. The sort then uses the partitioner recursively on the sub-arrays until the partitions are small enough to use shell sort on the final array.