Link to repo = https://github.com/scandum/wolfsort

The workload I chose is a benchmark of a sorting algorithm that the owner of the repository had made. The sorting algorithm in mind is called **wolfsort**, it is a hybrid sorting algorithm that combines properties of radix sort, quicksort and merge sort. The workload runs a benchmark that compares wolfsort to other hybrid sorting algorithms.

Downloading and running the workload

To download the repository, run the following command

```
git clone https://github.com/scandum/wolfsort
```

After cloning the repository, you can run and compile the benchmark as such:

```
cd wolfsort/src
gcc -O3 bench.c
./a.out
```

To compile it into a RISC-V binary run the following compile line

```
riscv64-linux-gnu-gcc -O3 -static bench.c -o bench.rv
```

The bash script I used to run the workload is the following code

```
ESESC_BIN=${1:-../main/esesc}

export ESESC_ReportFile="part2Report"
export ESESC_BenchName="./wolfsort/src/bench.rv"
export ESESC_DL1_core_bsize=8
if [ -f $ESESC_BIN ]; then
  $ESESC_BIN
else
  $ESESC_BenchName
fi
exit 0
```

Initial Results

The initial run of the benchmark showed an IPC of 0.41, and a total instruction count of 362,993,027 instructions, with a running time of 702 seconds. The full report given by esesc can be seen below