

# 情報学郡実験第 2 アセンブリ言語を用いたソートアルゴリズム の実装

1250372 三上柊

2022 年 11 月 4 日

## 第 1 章

# 本課題に関して

本課題で作成するアルゴリズムが満たすべき条件は、

- 指定された主記憶領域のダブルワード列を昇順に整列すること
- 整列対象の先頭番地が `ebx` に、ダブルワードの個数が `ecx` に格納されて `sort` に渡される
- `sort` は `ebx` が指す番地から始まる `ecx` 個のダブルワードからなる領域の中身を書き換える
- 整列対象のダブルワードは 0 以上  $2^{31}$  未満である
- ダブルワードの個数は 1 個以上 30 万個以下である
- `sort.s` には、サブルーチン `sort` の定義（及び、整列対象以外にこのサブルーチンが必要とするデータ領域等の定義）のみ記述すること。特に、`sort.s` の中でラベル `_start` を定義してはいけない。
- ラベル `sort` を `global` 宣言すること。
- `sort` は、標準出力に何も出力しないこと。
- `sort` の呼び出し前と復帰後で汎用レジスタの値が変化しないようにすること。

である。これを満たすよう作成したソートアルゴリズム（`sort.s`）に対して実験を行う。

## 第 2 章

# コンパイラを使用せず，アセンブリ，機械語のみでのソートアルゴリズムの記述

### 2.1 目的

この実験の目的は、「コンパイラを使用せず，アセンブリ，機械語のみでソートアルゴリズムを記述することが可能である」ことを示すことである。

### 2.2 方法

第 1 章に示した条件を満たすよう，高級言語で記述したものと同じ挙動をするようなソートアルゴリズムを記述する．また，使用したコンピュータ，実行コマンドは以下である．

Listing 2.1 使用したコンピュータ

---

```
$ uname -a
Linux KUT20VLIN-403 5.4.0-70-generic #78~18.04.1-Ubuntu SMP Sat Mar 20 14:10:07 UTC
2021 x86_64 x86_64 x86_64 GNU/Linux
```

---

print\_eax.s，test\_sort.s，sort.s に対して以下を行う．

Listing 2.2 実行コマンド

---

```
$ nasm -felf print_eax.s
$ nasm -felf test_sort.s
$ nasm -felf sort.s
$ ld -m elf_i386 print_eax.o test_sort.o sort.o
$ ./a.out
```

---

## 2.3 実験結果

a.out を実行した結果 , 右のような結果が得られた .	入力	1	3	5	7	9	2	4	6	8	0	1	2
この結果から , 正しく動作していると考えられる .	出力	0	1	1	2	2	3	4	5	6	7	8	9
	期待出力	0	1	1	2	2	3	4	5	6	7	8	9

## 2.4 考察

実験結果から , 「1 以上 30 万個以下の 0 以上  $2^{32}$  未満のダブルワード」に関してはアセンブリ言語でソートアルゴリズムが記述可能であると言える . この条件を満たさない場合のソートアルゴリズムも記述可能であるかは定かでない .