

アルゴリズム 2A 第 4 回レポート

61908697 佐々木良輔

1 結果

bubble sort, heap sort, counting sort について比較した。また計算時間は他のプロセスの CPU 使用率によって変化するため、コピー回数、比較回数で評価した。交換 1 回はコピー 3 回とした。

1.1 データ数による比較

rand10000.txt のデータの上から 7500, 5000, 2500 行を抽出し rand7500.txt, rand5000.txt, rand2500.txt を作成した。それぞれについて bubble sort, heap sort, counting sort を行い比較した。表 1 に結果を示す。

表 1 データ数による比較

行数	bubble sort		heap sort		counting sort	
	copy	compare	copy	compare	copy	compare
10000	1.50×10^8	2.49×10^7	1.74×10^5	2.35×10^5	2.00×10^4	0.00
7500	8.43×10^7	1.39×10^7	1.28×10^5	1.70×10^5	1.50×10^4	0.00
5000	3.74×10^7	6.34×10^6	8.21×10^4	1.08×10^5	1.00×10^4	0.00
2500	9.35×10^6	1.56×10^6	3.86×10^4	4.89×10^4	5.00×10^3	0.00

1.2 昇順, 降順, ランダム入力の比較

表 2 に結果を示す。

表 2 昇順, 降順, ランダム入力の比較

	bubble sort		heap sort		counting sort	
	copy	compare	copy	compare	copy	compare
descend	1.50×10^8	5.00×10^7	1.67×10^5	2.27×10^5	2.00×10^4	0.00
ascend	0.00	1.00×10^4	1.79×10^5	2.40×10^5	2.00×10^4	0.00
random	1.50×10^8	2.49×10^7	1.74×10^5	2.35×10^5	2.00×10^4	0.00

2 考察

2.1 データ数による比較

図 1, 図 2, 図 3 に bubble sort, heap sort, counting sort それぞれの計算量のデータ数依存性を示す. またそれぞれの図に $y = ax^2$, $y = ax \log x$, $y = ax$ でフィットした曲線を示している. これらから bubble sort の計算量は $y = ax^2$, heap sort の計算量は $y = ax \log x$, counting sort の計算量は $y = ax$ でよくフィットしていることがわかる.

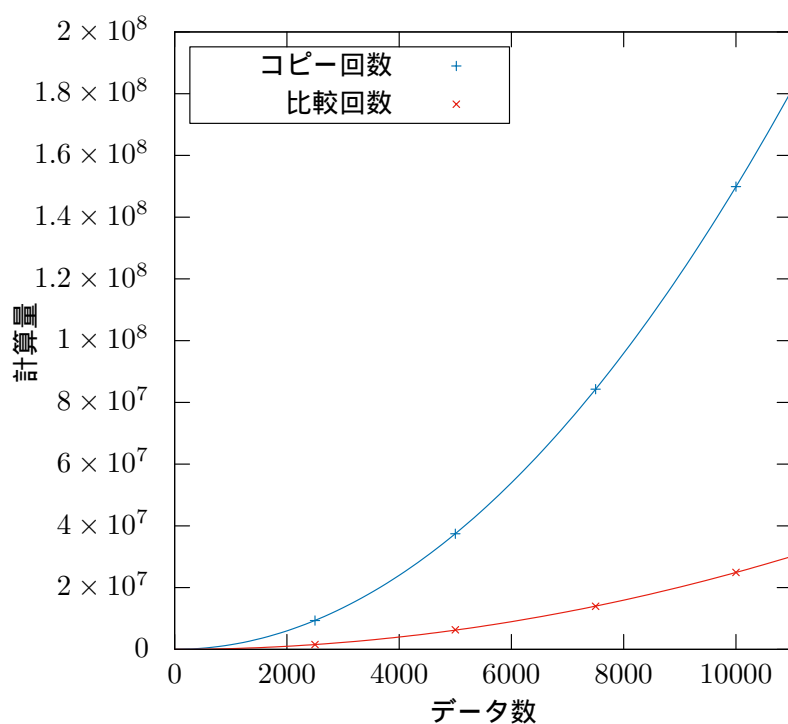


図 1 bubble sort のコピー, 比較回数

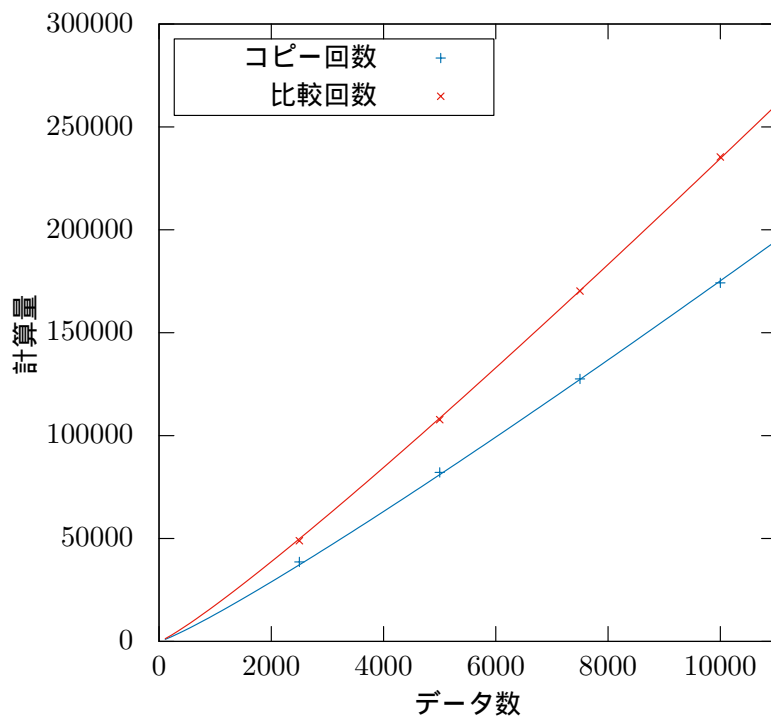


図 2 heap sort のコピー, 比較回数

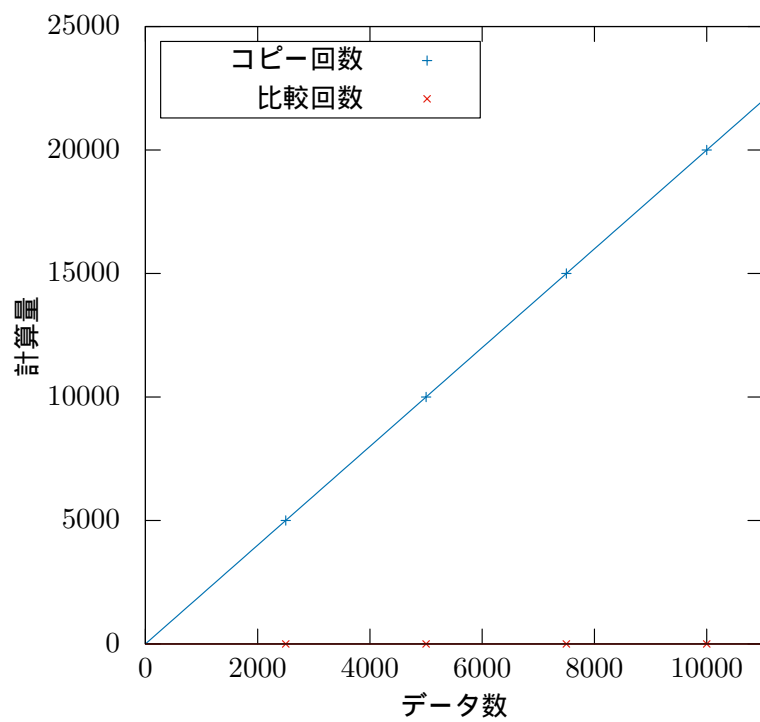


図 3 counting sort のコピー, 比較回数

2.2 昇順, 降順, ランダム入力の比較

2.2.1 bubble sort について

表 1 から descend, random の場合の計算量は 10^8 程度のオーダーになっているのに対し, ascend の場合は 10^4 程度のオーダーになっていることがわかる. これは ascend が既に sort されており, 交換が発生しなかった時点でループを抜けたためである. これは bubble sort の計算量が最悪で $O(n^2) \sim 10^8$, 平均で $O(n^2) \sim 10^8$, 最良で $O(n) \sim 10^4$ であることに整合する.

2.2.2 heap sort について

表 1 から descend, ascend, random 全ての場合において計算量は 10^5 程度のオーダーである. これは heap sort が全ての場合において $O(n \log_2 n) \sim 10^5$ となることに整合する.

2.2.3 counting sort について

表 1 から descend, ascend, random 全ての場合において計算量は 10^4 程度のオーダーである. これは counting sort が全ての場合において $O(n) \sim 10^4$ となることに整合する.