

# HOMework REPORT

## -Week 3 Sorting Algorithms-

### I. Thông tin nhóm:

- Thành viên: (nhóm 1 người)
  - o 24120072 – Lê Công Anh Khoa

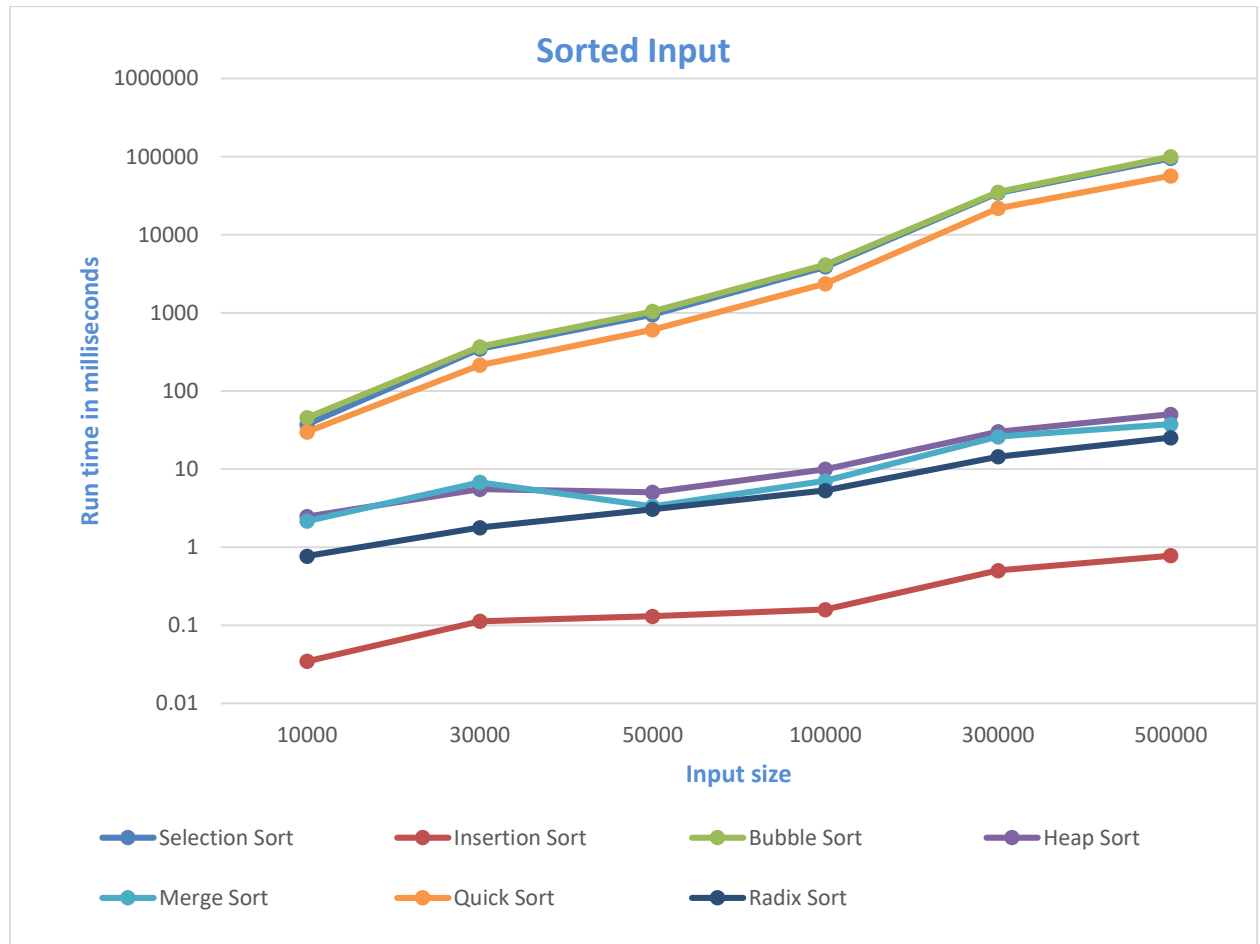
### II. Báo cáo về bài tập:

- Trong tuần vừa qua, em đã thực hiện tìm hiểu 7 thuật toán sắp xếp nằm trong set 1 được giao bao gồm: **Selection sort** - **Insertion sort** - **Bubble sort** - **Heap sort** - **Merge sort** - **Quick sort** - **Radix sort**. Và thu được kết quả bao gồm 8 sơ đồ bên dưới:
  - o 4 sơ đồ đường, mỗi sơ đồ là một thứ tự của dữ liệu đầu vào, ghi lại thời gian chạy của từng thuật toán (trục tung) ứng với kích thước dữ liệu đầu vào (trục hoành).
  - o 4 sơ đồ cột, mỗi sơ đồ là một thứ tự của dữ liệu đầu vào, ghi lại số lượng phép so sánh được dùng trong từng thuật toán (trục tung) ứng với kích thước dữ liệu đầu vào (trục hoành).
- Vì độ chênh lệch dữ liệu thu được là rất lớn giữa các thuật toán nên để hiển thị được sự khác nhau, em đã sử dụng chức năng logarithmic scale đồng thời đính kèm thêm bảng giá trị thu được bên dưới mỗi sơ đồ.
- Với thuật toán Quick sort, việc triển khai hàm phân hoạch như được học trong lớp lý thuyết (chọn phần tử pivot ở bên phải và 2 biến chạy đi từ trái qua phải để tách mảng) dẫn đến thời gian chạy trong trường hợp xấu nhất (mảng đã được sắp xếp) bị đẩy lên rất cao nên em đã thay đổi một chút về hàm phân hoạch, cụ thể là: chọn phần tử pivot ở bên trái và cho 2 biến chạy bắt đầu từ 2 đầu của mảng, di chuyển về phía đối diện, nếu biến chạy bên trái nhỏ hơn biến chạy bên phải và giá trị ở vị trí biến chạy bên trái lớn hơn biến chạy bên phải thì sẽ đổi chỗ 2 giá trị đó.

### III. Kết quả thu được:

#### A. Về thời gian:

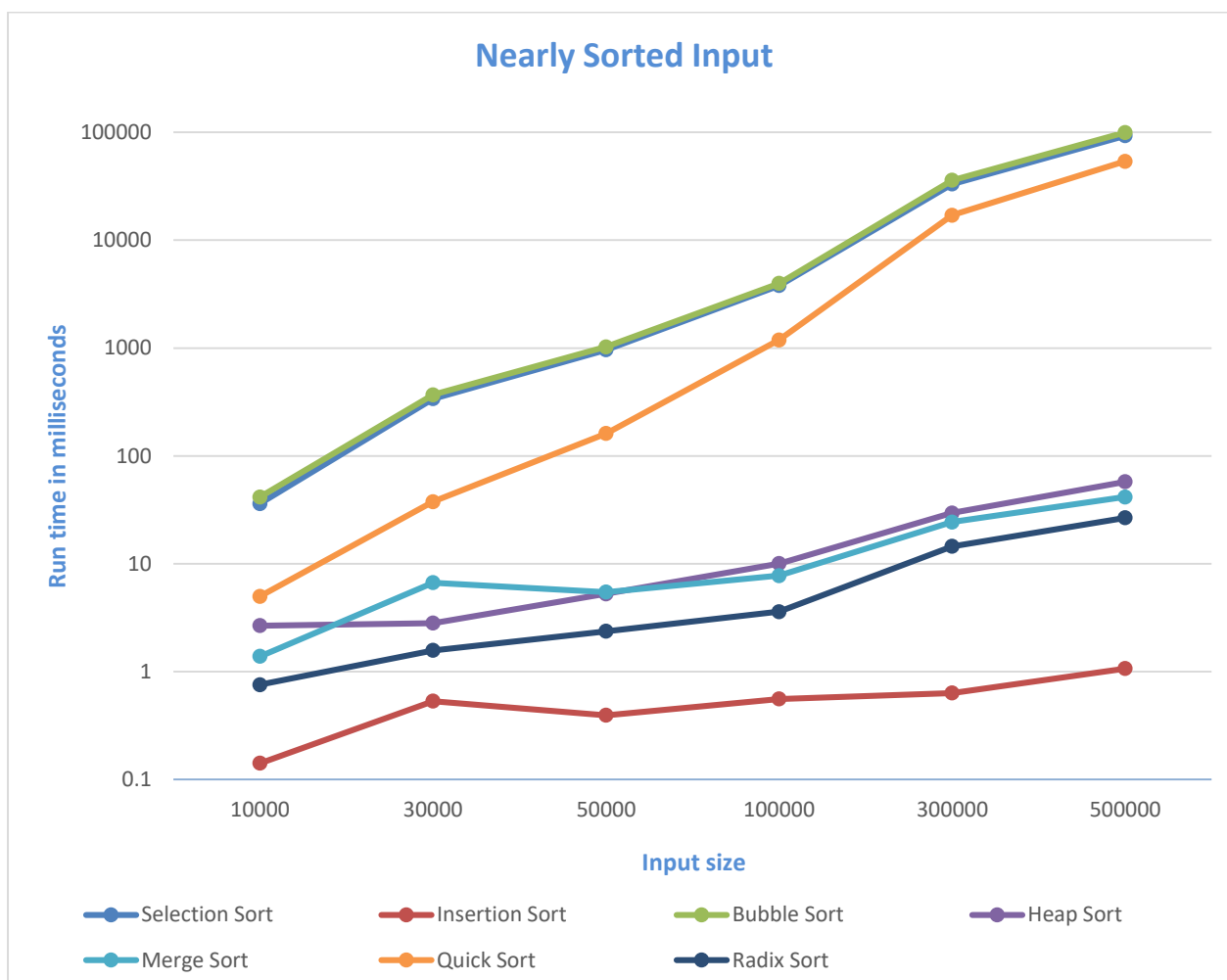
##### 1. Dữ liệu được sắp xếp sẵn:



<div>Số phần tử</div> <div>Thuật toán</div>	10000	30000	50000	100000	300000	500000
Selection Sort	37.3367	346.677	949.533	3885.28	34451.8	95376.6
Insertion Sort	0.0348	0.1132	0.1308	0.1595	0.505	0.7834
Bubble Sort	45.7848	370.002	1046.37	4135.98	35231	100336
Heap Sort	2.4792	5.5532	5.0629	10.0189	30.2012	50.5511
Merge Sort	2.1749	6.8023	3.3432	7.0921	26.0284	37.6756
Quick Sort	30.0761	214.388	609.395	2368.21	21931	57022.9
Radix Sort	0.7736	1.7917	3.0708	5.3375	14.4872	25.3316

Bảng 1: thời gian chạy thuật toán với dữ liệu được sắp xếp

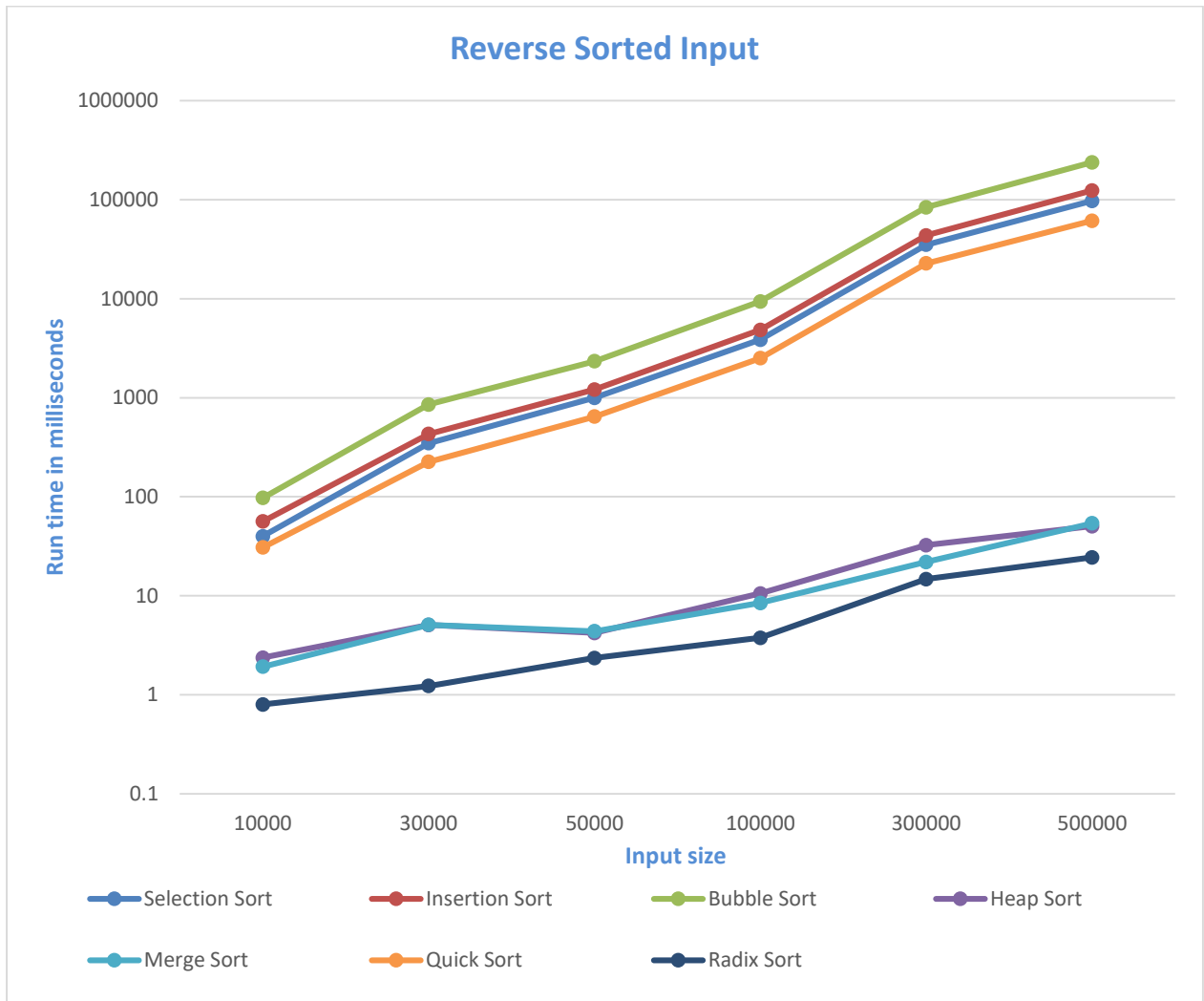
## 2. Dữ liệu gần như được sắp xếp:



Số phần tử Thuật toán	10000	30000	50000	100000	300000	500000
Selection Sort	36.1976	338.96	962.956	3793.64	33178.1	92994.8
Insertion Sort	0.1417	0.533	0.3948	0.5577	0.6337	1.0694
Bubble Sort	41.6034	369.396	1027.49	3981.8	35948.4	99737.3
Heap Sort	2.6759	2.8178	5.2772	10.0623	29.6477	57.5273
Merge Sort	1.3889	6.6737	5.4731	7.7592	24.3567	41.6656
Quick Sort	4.9972	37.7046	161.686	1188.29	17029.5	53872.7
Radix Sort	0.7582	1.5747	2.367	3.603	14.5264	26.6966

Bảng 2: thời gian chạy thuật toán với dữ liệu gần như sắp xếp

### 3. Dữ liệu bị sắp xếp đảo ngược:



Số phần tử Thuật toán	10000	30000	50000	100000	300000	500000
Selection Sort	39.8461	347.274	997.961	3850.05	35065.9	97158.7
Insertion Sort	56.2314	429.592	1208.2	4843.49	43504.3	123822
Bubble Sort	97.3708	852.912	2337.7	9381.32	83815.1	237905
Heap Sort	2.3716	5.0737	4.2168	10.5542	32.4103	50.3539
Merge Sort	1.9255	5.0977	4.3807	8.455	21.9713	53.9687
Quick Sort	30.7413	224.773	645.176	2507.93	22743.8	61271.6
Radix Sort	0.7977	1.2288	2.3531	3.7566	14.7148	24.3924

Bảng 3: thời gian chạy thuật toán với dữ liệu sắp xếp đảo ngược

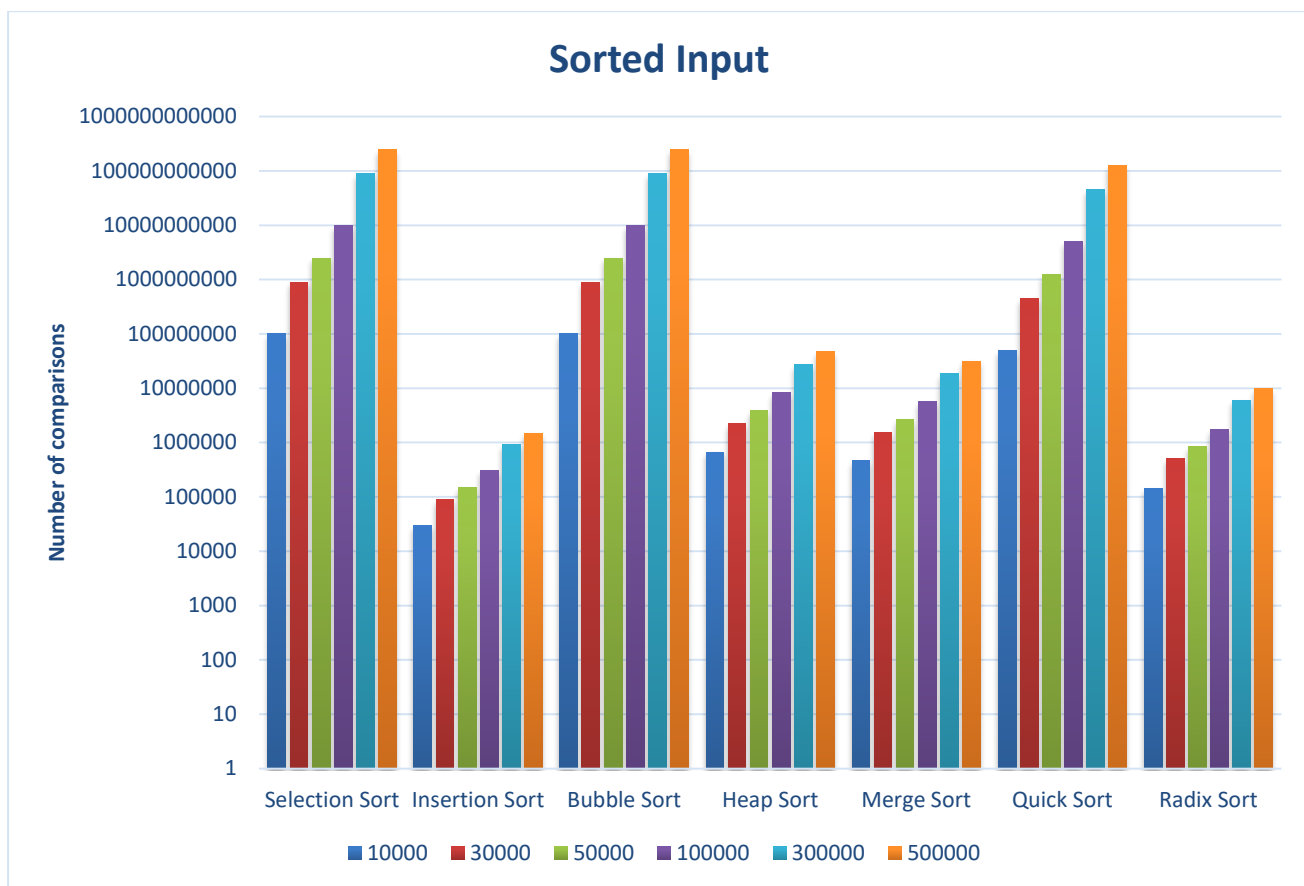
#### 4. Dữ liệu ngẫu nhiên:



Số phần tử Thuật toán	10000	30000	50000	100000	300000	500000
Selection Sort	42.66	337.506	959.506	3669.97	33969.5	95176.8
Insertion Sort	27.3595	220.625	598.561	2388.75	21136.5	61027.5
Bubble Sort	83.1743	1289.08	3889.36	16829.2	152710	428845
Heap Sort	3.0284	4.9947	10.7363	14.6552	43.3024	77.3653
Merge Sort	3.1398	7.342	7.3434	19.4142	44.912	85.6643
Quick Sort	0.8754	3.289	5.4144	8.7352	26.8095	43.356
Radix Sort	0.4077	1.4353	3.4608	4.0222	12.4581	20.7667

Bảng 4: thời gian chạy thuật toán với dữ liệu ngẫu nhiên

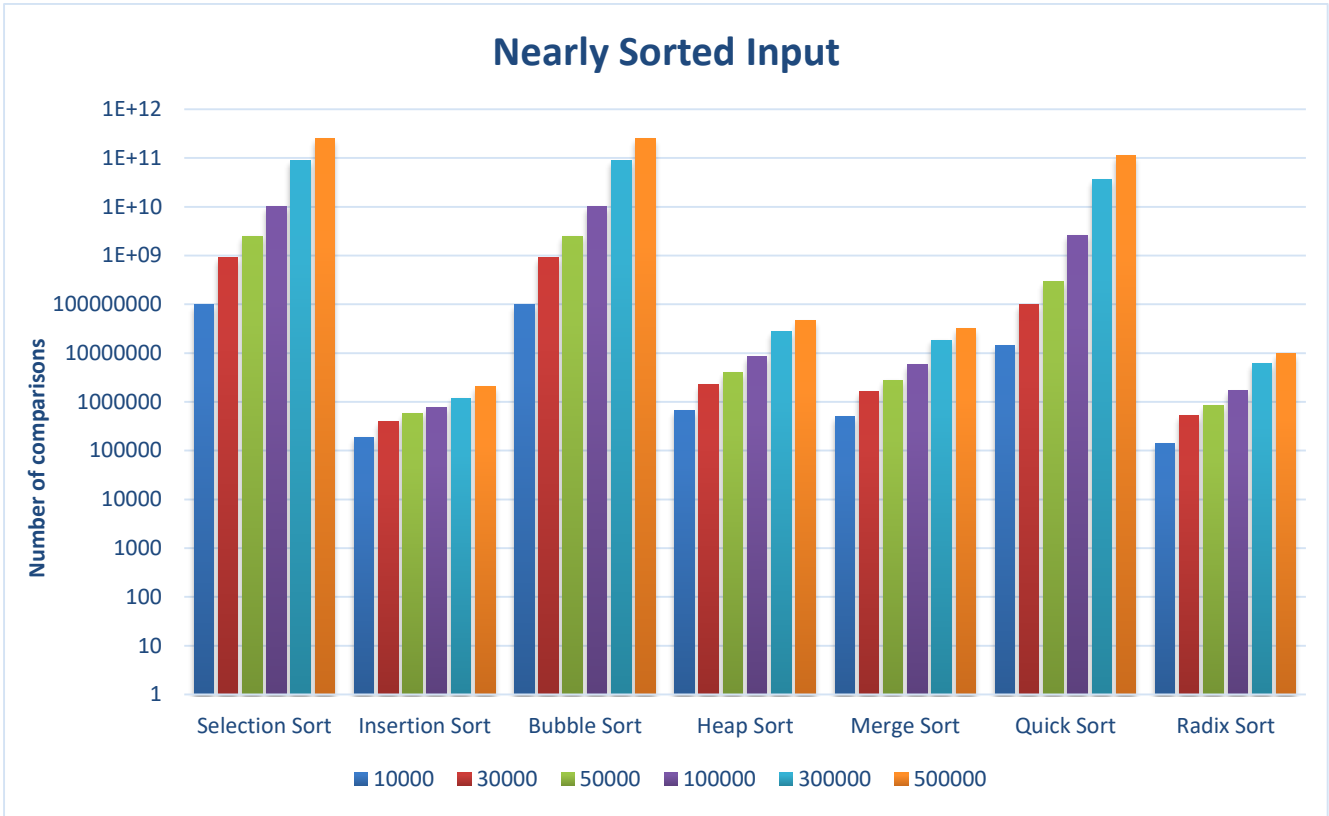
- B. Về số lượng phép so sánh:
- Dữ liệu được sắp xếp sẵn:



Số phần tử Thuật toán	10000	30000	50000	100000	300000	500000
Selection Sort	100010001	900030001	2500050001	10000100001	90000300001	250000500001
Insertion Sort	29998	89998	149998	299998	899998	1499998
Bubble Sort	100009999	900029999	2500049999	10000099999	90000299999	250000499999
Heap Sort	670329	2236648	3925351	8365080	27413230	47404886
Merge Sort	465243	1529915	2672827	5645659	18345947	31517851
Quick Sort	50044996	450134996	1250224996	5000449996	45001349996	125002249996
Radix Sort	140056	510070	850070	1700070	6000084	10000084

Bảng 5: Số lượng phép so sánh trong thuật toán với dữ liệu sắp xếp sẵn

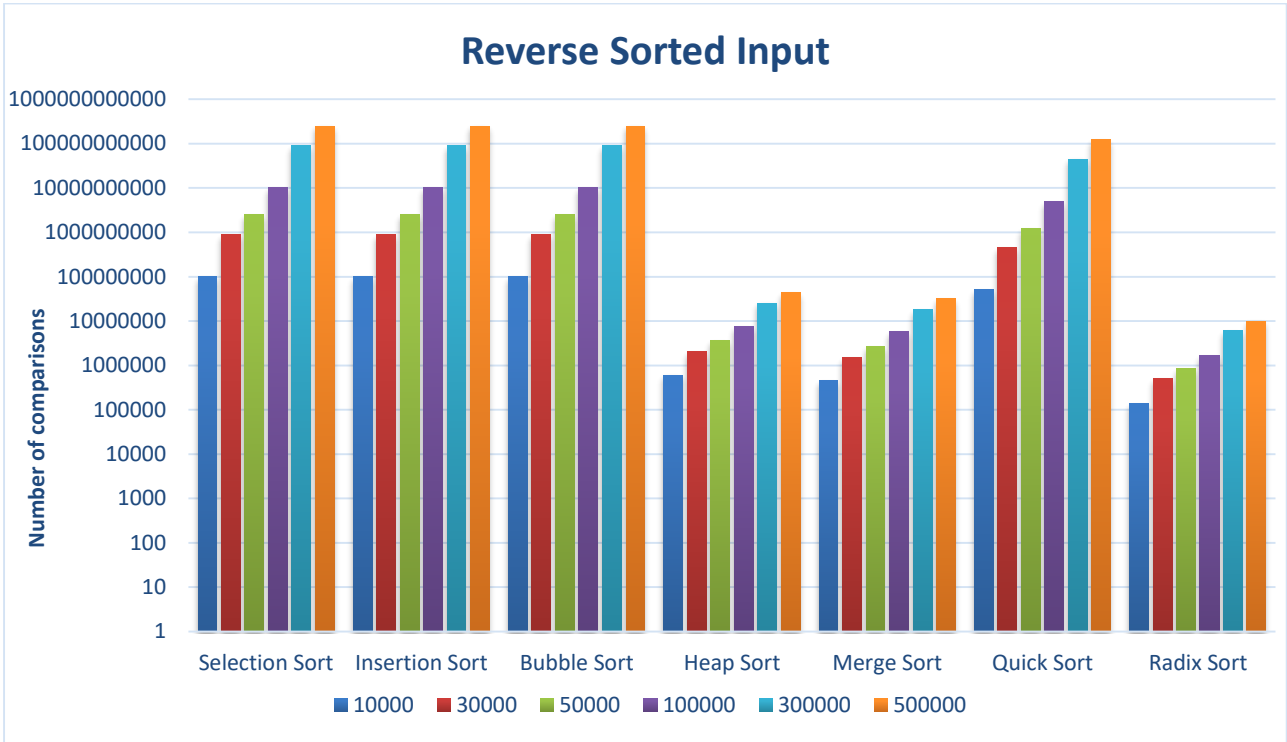
2. Dữ liệu gần như được sắp xếp:



<div>Số phần tử</div> <div>Thuật toán</div>	10000	30000	50000	100000	300000	500000
Selection Sort	100010001	900030001	2500050001	10000100001	90000300001	250000500001
Insertion Sort	187882	395630	582478	749366	1205198	2048638
Bubble Sort	100009999	900029999	2500049999	10000099999	90000299999	250000499999
Heap Sort	669997	2236522	3925390	8364752	27413150	47405175
Merge Sort	491485	1599946	2785460	5739107	18441874	31641059
Quick Sort	14218495	96894569	287930135	2618348100	36675003406	111397419320
Radix Sort	140056	510070	850070	1700070	6000084	10000084

Bảng 6: Số lượng phép so sánh trong thuật toán với dữ liệu gần như sắp xếp

3. Dữ liệu bị sắp xếp đảo ngược:

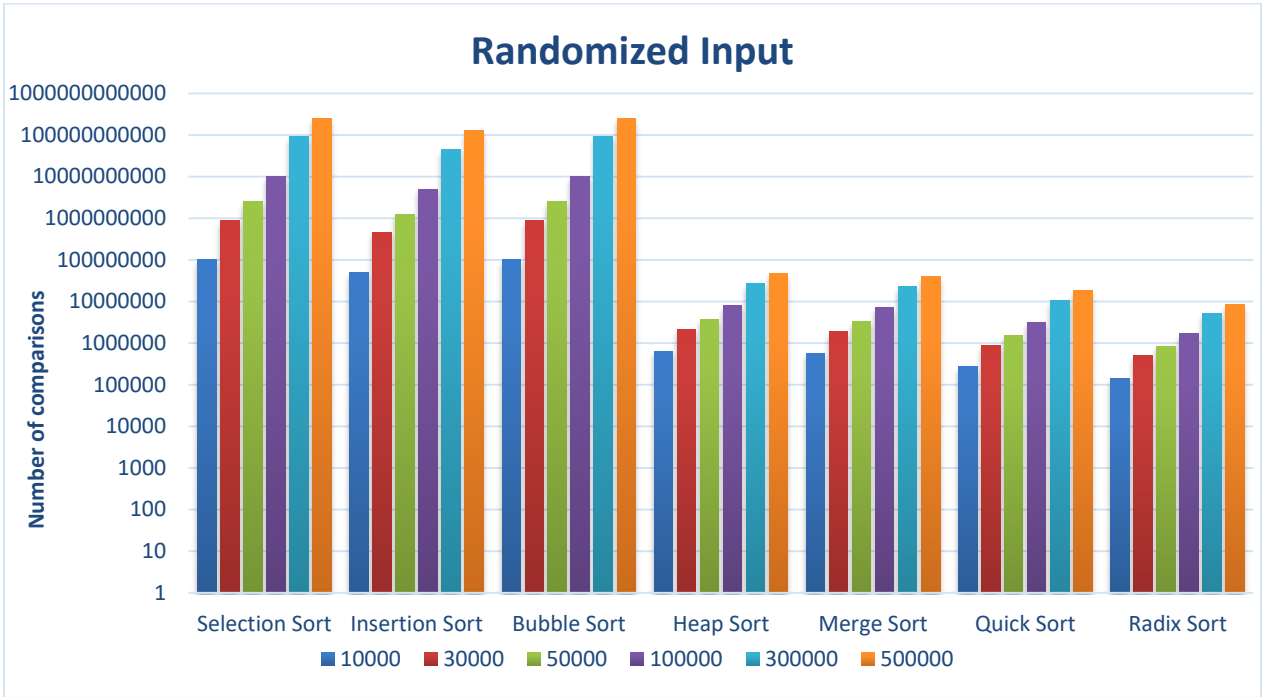


<div>Số phần tử</div> <div>Thuật toán</div>	10000	30000	50000	100000	300000	500000
Selection Sort	100010001	900030001	2500050001	10000100001	90000300001	250000500001
Insertion Sort	100009999	900029999	2500049999	10000099999	90000299999	250000499999
Bubble Sort	100009999	900029999	2500049999	10000099999	90000299999	250000499999
Heap Sort	606771	2063324	3612724	7718943	25569379	44483348
Merge Sort	466442	1543466	2683946	5667898	18408314	31836410
Quick Sort	50054996	450164996	1250274996	5000549996	45001649996	125002749996
Radix Sort	140056	510070	850070	1700070	6000084	10000084

Bảng 7: Số lượng phép so sánh trong thuật toán với dữ liệu gần như sắp xếp



4. Dữ liệu ngẫu nhiên:



<div>Số phần tử</div> <div>Thuật toán</div>	10000	30000	50000	100000	300000	500000
Selection Sort	100010001	900030001	2500050001	10000100001	90000300001	250000500001
Insertion Sort	50241229	450779787	1246939373	4999149212	45100800313	124945686810
Bubble Sort	100009999	900029999	2500049999	10000099999	90000299999	250000499999
Heap Sort	637752	2150135	3772029	8043305	26489799	45970891
Merge Sort	573621	1907325	3333775	7066099	23082129	39881502
Quick Sort	275219	880571	1558122	3204432	10509792	18087823
Radix Sort	140056	510070	850070	1700070	5100070	8500070

Bảng 8: Số lượng phép so sánh trong thuật toán với dữ liệu gần như sắp xếp