

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний
інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки

Кафедра ІІІ

Звіт

з лабораторної роботи № 1 з дисципліни
«Алгоритми та структури даних 2. Структури даних»

„Проектування і аналіз алгоритмів внутрішнього сортування”

Виконав(ла)

ІІ-12, Кириченко Владислав Сергійович
(шифр, прізвище, ім'я, по батькові)

Перевірів

Халус Олена Андріївна _____
(прізвище, ім'я, по батькові)

Київ 2022

ЗМІСТ

1	МЕТА ЛАБОРАТОРНОЇ РОБОТИ.....	3
2	ЗАВДАННЯ.....	4
3	ВИКОНАННЯ.....	5
3.1	АНАЛІЗ АЛГОРИТМУ НА ВІДПОВІДНІСТЬ ВЛАСТИВОСТЯМ.....	5
3.2	ПСЕВДОКОД АЛГОРИТМУ.....	6
3.3	АНАЛІЗ ЧАСОВОЇ СКЛАДНОСТІ.....	6
3.4	ПРОГРАМНА РЕАЛІЗАЦІЯ АЛГОРИТМУ.....	6
3.4.1	<i>Вихідний код.....</i>	<i>7</i>
3.4.2	<i>Приклад роботи.....</i>	<i>8</i>
3.5	ТЕСТУВАННЯ АЛГОРИТМУ.....	10
3.5.1	<i>Часові характеристики оцінювання.....</i>	<i>10</i>
3.5.2	<i>Графіки залежності часових характеристик оцінювання від розмірності масиву.....</i>	<i>12</i>
	ВИСНОВОК.....	21
	КРИТЕРІЇ ОЦІНЮВАННЯ.....	22

1 МЕТА ЛАБОРАТОРНОЇ РОБОТИ

Мета роботи – вивчити основні методи аналізу обчислювальної складності алгоритмів внутрішнього сортування і оцінити поріг їх ефективності.

2 ЗАВДАННЯ

Виконати аналіз алгоритму внутрішнього сортування на відповідність наступним властивостям (таблиця 2.1):

- стійкість;
- «природність» поведінки (Adaptability);
- базуються на порівняннях;
- необхідність додаткової пам'яті (об'єму);
- необхідність в знаннях про структуру даних.

Записати алгоритм внутрішнього сортування за допомогою псевдокоду (чи іншого способу по вибору).

Провести аналіз часової складності в гіршому, кращому і середньому випадках та записати часову складність в асимптотичних оцінках.

Виконати програмну реалізацію алгоритму на будь-якій мові програмування з фіксацією часових характеристик оцінювання (кількість порівнянь, кількість перестановок, глибина рекурсивного поглиблення та інше в залежності від алгоритму).

Провести ряд випробувань алгоритму на масивах різної розмірності (10, 100, 1000, 5000, 10000, 20000, 50000 елементів) і різних наборів вхідних даних (впорядкований масив, зворотно упорядкований масив, масив випадкових чисел) і побудувати графіки залежності часових характеристик оцінювання від розмірності масиву, нанести на графік асимптотичну оцінку гіршого і кращого випадків для порівняння.

Зробити порівняльний аналіз двох алгоритмів.

Зробити узагальнений висновок з лабораторної роботи.

Таблиця 2.1 – Варіанти алгоритмів

№	Алгоритм сортування
1	Сортування бульбашкою
2	Сортування гребінцем («розчіскою»)

ВИКОНАННЯ

БУЛЬБАШКА

2.1 Аналіз алгоритму на відповідність властивостям

Аналіз алгоритму сортування бульбашкою на відповідність властивостям наведено в таблиці 3.1.

Таблиця 3.1 – Аналіз алгоритму на відповідність властивостям

Властивість	Сортування бульбашкою
Стійкість	Алгоритм є стійким
«Природність» поведінки (Adaptability)	Алгоритм не є природнім
Базуються на порівняннях	Алгоритм базується на порівняннях
Необхідність в додатковій пам'яті (об'єм)	$O(1)$
Необхідність в знаннях про структури даних	Присутня

2.2 Псевдокод алгоритму

```
for i ← 1 to length[arr]
  do
    for j ← 2 to length[arr] - i
      do
        if arr[j] < arr[j-1]
          then
            temp ← arr[j]
            arr[j] ← arr[j-1]
            arr[j-1] ← temp
          end if
        end if
      end for
    end for
```

2.3 Аналіз часової складності

for i ← 1 to length[arr]	<i>n-pazib</i>
<i>do</i>	
<i>for</i> j ← 2 to length[arr] - i	$\sum_{i=1}^{n-1} (n - i)$
do	
if arr[j] < arr[j-1]	$\sum_{i=1}^{n-1} (n - i)$
<i>then</i>	
temp ← arr[j]	$0 - \sum_{i=1}^{n-1} (n - i)$
arr[j] ← arr[j-1]	$0 - \sum_{i=1}^{n-1} (n - i)$
arr[j-1] ← temp	$0 - \sum_{i=1}^{n-1} (n - i)$
<i>end if</i>	
end if	
<i>end for</i>	
<i>end for</i>	

Найліпший випадок - $\Omega(n^2)$

Найгірший випадок - $O(n^2)$

Найгірший випадок для перестановок = Найгірший випадок загальний

ДОВЕДЕННЯ

The image shows a handwritten derivation on grid paper. The first part shows the summation of comparisons:
$$T(n) = \sum_{i=1}^{n-1} (n-i) = \frac{n-1+1}{2} (n-1) = \frac{n}{2} (n-1) = \frac{n^2}{2} - \frac{n}{2}$$
 The second part shows the final complexity:
$$T_n = \cancel{O(n^2)} O(n^2)$$

2.4 Програмна реалізація алгоритму

2.4.1 Вихідний код

```
def bubble_sort(arr):  
    length = len(arr)  
    for i in range(0, length):  
        for j in range(1, length-i):  
            if arr[j] < arr[j-1]:  
                arr[j], arr[j-1] = arr[j-1], arr[j]
```

2.4.2 Приклад роботи

На рисунках 3.1 і 3.2 показані приклади роботи програми сортування масивів на 100 і 1000 елементів відповідно.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/powershell

PS E:\workplace\kpi\asd\2semester\lab1> & 'C:\Users\advel\AppData\Local\Microsoft\WindowsApps\python3.9.exe' 'c:\Users\advel\.vscode\extensions\ms-python.python-2022.0.1814523869\pythonFiles\lib\python\debugpy\launcher' '61555' '--' 'e:\workplace\kpi\asd\2semester\lab1\main.py'
Enter array size: 100
Initial array:
[60, 39, 44, 10, 28, 66, 79, 55, 36, 22, 56, 77, 85, 16, 52, 80, 88, 46, 50, 86, 0, 31, 20, 23, 11, 63, 98, 70, 47, 64, 72, 41, 53, 6, 75, 33, 2, 1, 26, 87, 84, 91, 57, 94, 65, 4, 40, 74, 14, 7, 49, 58, 92, 19, 69, 27, 21, 9, 13, 15, 89, 34, 61, 43, 95, 97, 59, 30, 25, 24, 78, 81, 83, 38, 99, 29, 5, 71, 62, 96, 51, 18, 8, 93, 32, 76, 90, 82, 45, 54, 12, 48, 68, 73, 37, 3, 67, 17, 42, 35]
Sorted array:
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99]
PS E:\workplace\kpi\asd\2semester\lab1>
```

Рисунок 3.1 – Сортування масиву на 100 елементів


```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/powershell

PS E:\workplace\kpi\asd\2semester\lab1> & 'C:\Users\advel\AppData\Local\Microsoft\WindowsApps\python3.9.exe' 'c:\Users\advel\.vscode\extensions\ms-python.python-2022.0.1814523869\pythonFiles\lib\python\debugpy\launcher' '61634' '--' 'e:\workplace\kpi\asd\2semester\lab1\main.py'
Enter array size: 1000
Initial array:
[790, 293, 997, 451, 182, 799, 757, 400, 594, 956, 900, 501, 98, 553, 610, 589, 503, 573, 522, 66, 768, 708, 41, 36, 289, 354, 941, 266, 385,
695, 958, 409, 389, 325, 990, 704, 540, 446, 374, 137, 766, 676, 546, 593, 196, 12, 105, 81, 891, 326, 858, 156, 277, 934, 2, 713, 845, 669,
960, 6, 494, 549, 146, 740, 987, 454, 904, 307, 578, 258, 490, 909, 545, 531, 638, 323, 149, 645, 913, 216, 961, 276, 438, 597, 305, 397, 780,
577, 971, 574, 809, 527, 502, 214, 278, 411, 153, 35, 448, 230, 859, 517, 221, 681, 259, 832, 998, 391, 731, 529, 190, 735, 738, 769, 140, 41
8, 745, 65, 296, 363, 370, 687, 386, 61, 348, 228, 884, 996, 723, 855, 614, 251, 838, 29, 435, 954, 321, 319, 406, 158, 737, 90, 802, 34, 515,
548, 513, 480, 829, 142, 814, 542, 256, 508, 426, 562, 619, 284, 309, 163, 108, 658, 425, 103, 718, 477, 94, 994, 428, 734, 950, 582, 707, 37
3, 250, 783, 359, 202, 559, 285, 246, 591, 164, 785, 948, 926, 257, 0, 145, 401, 805, 728, 439, 927, 662, 774, 898, 746, 86, 984, 127, 605, 53
5, 777, 683, 627, 771, 434, 862, 380, 197, 4, 459, 355, 143, 643, 180, 974, 534, 516, 792, 725, 168, 694, 576, 111, 833, 652, 579, 341, 244, 7
6, 112, 249, 201, 147, 96, 824, 333, 856, 547, 227, 905, 671, 399, 23, 730, 888, 609, 290, 860, 945, 930, 148, 260, 136, 615, 457, 726, 592, 6
33, 688, 890, 58, 788, 844, 709, 634, 91, 124, 189, 872, 753, 358, 524, 523, 372, 815, 19, 402, 977, 907, 581, 45, 11, 747, 104, 353, 772, 416
, 419, 808, 689, 150, 693, 484, 369, 647, 918, 176, 13, 519, 791, 317, 327, 346, 839, 536, 915, 642, 120, 56, 133, 967, 119, 165, 337, 413, 72
7, 394, 910, 10, 338, 93, 667, 43, 843, 241, 449, 268, 580, 754, 526, 959, 70, 166, 955, 611, 427, 572, 211, 382, 420, 331, 24, 152, 795, 287,
75, 882, 337, 255, 33, 887, 128, 719, 952, 118, 27, 79, 473, 209, 564, 680, 361, 436, 460, 476, 507, 668, 491, 303, 410, 584, 651, 699, 846,
863, 40, 281, 53, 412, 866, 697, 770, 239, 482, 729, 475, 835, 101, 674, 110, 187, 39, 935, 47, 506, 939, 195, 233, 901, 632, 493, 179, 654, 2
5, 640, 558, 452, 962, 942, 922, 199, 92, 132, 865, 999, 329, 95, 483, 665, 217, 282, 982, 637, 274, 424, 973, 869, 194, 188, 107, 742, 83, 45
0, 407, 664, 871, 532, 739, 696, 899, 620, 555, 339, 48, 928, 979, 585, 923, 336, 937, 54, 981, 612, 340, 892, 656, 248, 556, 870, 181, 541, 7
16, 798, 192, 625, 496, 781, 629, 943, 390, 240, 782, 151, 561, 635, 299, 936, 657, 114, 421, 462, 219, 711, 31, 441, 200, 818, 852, 335, 505,
99, 443, 320, 896, 831, 760, 5, 970, 481, 878, 154, 512, 682, 14, 242, 946, 883, 575, 847, 455, 968, 193, 308, 430, 750, 292, 604, 678, 804,
26, 488, 776, 902, 383, 710, 312, 743, 440, 67, 602, 64, 692, 828, 499, 919, 600, 470, 721, 263, 702, 539, 938, 485, 7, 344, 203, 162, 318, 41
5, 20, 422, 231, 125, 87, 940, 306, 504, 931, 315, 673, 857, 983, 821, 72, 63, 715, 670, 172, 552, 393, 469, 357, 851, 134, 916, 787, 920, 758
, 367, 613, 471, 895, 778, 650, 784, 511, 100, 262, 775, 570, 903, 295, 322, 300, 759, 924, 749, 384, 269, 316, 993, 183, 267, 992, 533, 55, 8
73, 206, 991, 261, 914, 776, 569, 820, 342, 186, 392, 378, 423, 717, 88, 52, 212, 467, 827, 700, 510, 583, 379, 62, 995, 264, 714, 17, 116, 21
, 184, 310, 288, 712, 789, 405, 97, 966, 362, 543, 889, 37, 867, 661, 528, 297, 841, 366, 238, 222, 495, 442, 986, 9, 603, 617, 414, 347, 169,
911, 350, 587, 590, 755, 330, 724, 885, 621, 666, 601, 283, 18, 74, 207, 631, 472, 530, 198, 744, 345, 497, 218, 840, 82, 161, 880, 698, 298,
794, 608, 586, 343, 701, 571, 138, 653, 756, 314, 68, 272, 811, 672, 741, 360, 588, 566, 925, 975, 131, 403, 431, 538, 554, 550, 812, 463, 45
8, 365, 816, 492, 122, 803, 659, 881, 69, 663, 50, 16, 761, 822, 985, 121, 853, 46, 874, 328, 42, 135, 868, 78, 171, 466, 44, 837, 825, 313, 3
0, 877, 376, 906, 751, 551, 270, 461, 797, 432, 32, 468, 177, 514, 691, 141, 969, 628, 849, 417, 807, 876, 826, 875, 624, 174, 445, 139, 208,
622, 703, 830, 800, 474, 395, 8, 796, 879, 598, 765, 265, 178, 630, 568, 660, 836, 155, 273, 677, 486, 951, 921, 271, 819, 560, 245, 226, 489,
170, 456, 252, 113, 286, 59, 129, 854, 224, 690, 655, 213, 988, 185, 706, 356, 73, 160, 616, 130, 817, 447, 520, 467, 396, 368, 686, 464, 371
, 223, 15, 933, 521, 234, 243, 279, 748, 157, 648, 599, 763, 275, 893, 679, 644, 429, 908, 685, 364, 167, 733, 636, 388, 210, 89, 953, 232, 50
9, 932, 563, 331, 618, 351, 767, 324, 175, 813, 500, 117, 294, 375, 51, 229, 349, 912, 204, 806, 834, 453, 929, 894, 675, 917, 801, 518, 334,
77, 109, 487, 398, 60, 596, 897, 498, 606, 254, 810, 444, 479, 302, 623, 102, 989, 28, 732, 38, 595, 705, 3, 964, 123, 722, 972, 957, 762, 944
, 22, 478, 779, 225, 565, 106, 793, 352, 173, 965, 861, 978, 525, 607, 764, 684, 236, 544, 159, 291, 408, 842, 84, 301, 220, 639, 144, 980, 56
7, 864, 641, 752, 71, 57, 720, 205, 247, 49, 823, 304, 626, 253, 381, 557, 646, 280, 649, 387, 433, 736, 126, 886, 332, 377, 235, 786, 80, 963
, 437, 237, 465, 850, 404, 949, 848, 1, 85, 115, 191, 215, 773]
Sorted array:
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37,
38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73
, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107,
108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135,
136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164
, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 1
93, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221,
222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250
, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278,
279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307
, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336,
337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364,
365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393
, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421,
422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450
, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507,
508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564,
565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593,
594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650,
651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707,
708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736,
737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793,
794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850,
851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936,
937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993,
994, 995, 996, 997, 998, 999]
PS E:\workplace\kpi\asd\2semester\lab1>

```

Рисунок 3.2 – Сортвання масиву на 1000 елементів

2.5 Тестування алгоритму

2.5.1 Часові характеристики оцінювання

В таблиці 3.2 наведені характеристики оцінювання числа порівнянь і числа перестановок алгоритму сортування бульбашки для масивів різної розмірності, коли масив містить упорядковану послідовність елементів.

Таблиця 3.2 – Характеристики оцінювання алгоритму сортування бульбашки для упорядкованої послідовності елементів у масиві

Розмірність масиву	Число порівнянь	Число перестановок
10	45	0
100	4950	0
1000	499500	0
5000	12497500	0
10000	49995000	0
20000	199990000	0
50000	1249975000	0

В таблиці 3.3 наведені характеристики оцінювання числа порівнянь і числа перестановок алгоритму сортування бульбашки для масивів різної розмірності, коли масиви містять зворотно упорядковану послідовність елементів.

Таблиця 3.3 – Характеристики оцінювання алгоритму сортування бульбашки для зворотно упорядкованої послідовності елементів у масиві.

Розмірність масиву	Число порівнянь	Число перестановок
10	45	45
100	4950	4950
1000	499500	499500
5000	12497500	12497500
10000	49995000	49995000
20000	199990000	199990000
50000	1249975000	1249975000

У таблиці 3.4 наведені характеристики оцінювання числа порівнянь і числа перестановок алгоритму сортування бульбашки для масивів різної розмірності, масиви містять випадкову послідовність елементів.

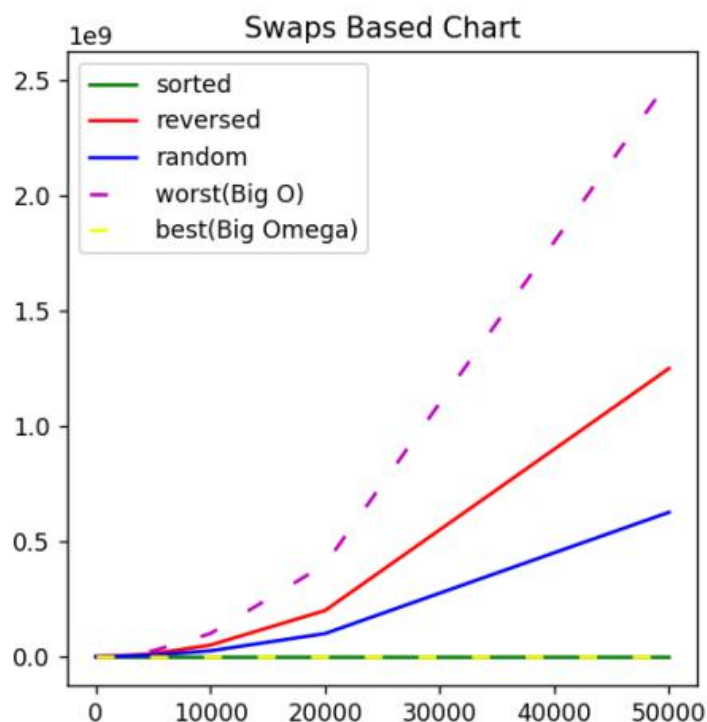
Таблиця 3.4 – Характеристика оцінювання алгоритму сортування бульбашки для випадкової послідовності елементів у масиві.

Розмірність масиву	Число порівнянь	Число перестановок
10	45	26
100	4950	2642
1000	499500	251401
5000	12497500	6195007
10000	49995000	25055888
20000	199990000	99971380
50000	1249975000	626023361

2.5.2 Графіки залежності часових характеристик оцінювання від розмірності масиву

На рисунку 3.3 показані графіки залежності часових характеристик оцінювання від розмірності масиву для випадків, коли масиви містять упорядковану послідовність елементів (зелений графік), коли масиви містять зворотно упорядковану послідовність елементів (червоний графік), коли масиви містять випадкову послідовність елементів (синій графік), також показані асимптотичні оцінки гіршого (фіолетовий графік) і кращого (жовтий графік) випадків для порівняння.

Рисунок 3.3 – Графіки залежності часових характеристик оцінювання



3 ВИКОНАННЯ ГРЕБІНЕЦЬ

3.1 Аналіз алгоритму на відповідність властивостям

Аналіз алгоритму сортування бульбашкою на відповідність властивостям наведено в таблиці 3.1.

Таблиця 3.1 – Аналіз алгоритму на відповідність властивостям

Властивість	Сортування гребінцем
Стійкість	Алгоритм не є стійким
«Природність» поведінки (Adaptability)	Алгоритм не є природнім
Базуються на порівняннях	Алгоритм базується на порівняннях
Необхідність в додатковій пам'яті (об'єм)	$O(1)$
Необхідність в знаннях про структури даних	Присутня

3.2 Псевдокод алгоритму

```
swapped  $\leftarrow$  False
length  $\leftarrow$  length[arr]
gap  $\leftarrow$  length mod 1.247

while gap > 1 || swapped == True
  do
    swapped = False
    for i  $\leftarrow$  1 to length - gap
      do
        if arr[i] > arr[i+gap]
          then
            temp  $\leftarrow$  arr[i]
            arr[i]  $\leftarrow$  arr[i+gap]
            arr[i+gap]  $\leftarrow$  temp
        If end
      if gap > 1
        then
          gap  $\leftarrow$  gap mod 1.247
        if end
    while end
```

3.3 Аналіз часової складності

Найліпший випадок - $\Omega(n \times \log_2 n)$

Найгірший випадок - $O(n \times \log_2 n) \mid O(n^{**2})$

Найгірший випадок для перестановок = загальний найгірший випадок

ДОВЕДЕННЯ

$$\begin{aligned}
 T(n) &= \sum_{i=0}^{\log_{1,3} n} \left(n - \frac{n}{1,3^i} \right) = \sum_{i=0}^{\log_{1,3} n} n - n \sum_{i=0}^{\log_{1,3} n} \frac{1}{1,3^i} \\
 &= n \cdot \log_{1,3} n - n \cdot \frac{1 - \left(\frac{1}{1,3}\right)^{\log_{1,3} n}}{1 - \frac{1}{1,3}} \\
 &= n \cdot \log_{1,3} n - n \cdot \frac{1 - \left(\frac{1}{1,3}\right)^{\log_{1,3} n}}{0,3} = n \cdot \log_{1,3} n - \frac{n}{0,3} + \frac{n}{0,3 \cdot (1,3)^{\log_{1,3} n}} \\
 &= n \cdot \log_{1,3} n - \frac{n}{0,3} + \frac{1}{0,3} \\
 \underline{T(n)} &= \underline{O(n \cdot \log_2 n)}
 \end{aligned}$$

3.4 Програмна реалізація алгоритму

3.4.1 Вихідний код

```
def comb_sort(arr):
    swapped = False
    length = len(arr)
    gap = int(length/1.247)

    while gap > 1 or swapped == True:
        swapped = False
        for i in range(length - gap):
            if arr[i] > arr[i+gap]:
                arr[i], arr[i+gap] = arr[i+gap], arr[i]
                swapped = True
        if gap > 1 :
            gap = int(gap/1.247)
```

3.4.2 Приклад роботи

На рисунках 3.1 і 3.2 показані приклади роботи програми сортування масивів на 100 і 1000 елементів відповідно.

```
INITIAL ARRAY:
[94, 86, 27, 34, 57, 19, 25, 80, 83, 62, 40, 26, 74, 11, 42, 99, 78, 20, 79, 66, 46, 9, 52, 75, 21, 5, 58, 45, 90, 51, 38, 70, 43, 60, 36, 56, 87, 1, 13, 44, 65,
32, 97, 84, 3, 30, 47, 0, 8, 39, 98, 29, 64, 16, 41, 96, 71, 7, 28, 18, 15, 82, 55, 63, 61, 35, 37, 24, 17, 92, 76, 53, 72, 81, 91, 67, 48, 10, 95, 77, 22, 14, 31
, 69, 12, 23, 93, 50, 59, 54, 4, 6, 73, 68, 88, 85, 33, 49, 2, 89]
SORTED ARRAY:
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42,
43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 8
3, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99]
```

Рисунок 3.1 – Сортування масиву на 100 елементів

SORTED ARRAY:
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999]

INITIAL ARRAY:
[848, 344, 471, 69, 320, 395, 892, 575, 363, 754, 652, 456, 418, 94, 275, 317, 573, 515, 681, 614, 15, 842, 314, 859, 378, 435, 733, 431, 665, 428, 218, 274, 501, 843, 565, 1, 43, 901, 525, 785, 173, 852, 819, 171, 364, 379, 869, 817, 313, 912, 35, 564, 541, 974, 472, 969, 153, 615, 278, 616, 140, 397, 461, 302, 392, 964, 954, 641, 905, 921, 165, 831, 514, 115, 440, 371, 600, 224, 624, 348, 749, 256, 758, 244, 830, 495, 287, 396, 157, 292, 460, 430, 151, 308, 332, 544, 778, 77, 991, 873, 594, 8, 436, 755, 277, 711, 118, 96, 868, 271, 527, 990, 555, 182, 703, 370, 185, 965, 757, 828, 81, 55, 156, 78, 485, 579, 975, 588, 721, 276, 765, 116, 1, 86, 98, 123, 598, 949, 608, 164, 311, 443, 413, 679, 107, 760, 774, 732, 110, 303, 467, 499, 533, 995, 896, 694, 87, 666, 850, 640, 548, 937, 769, 626, 136, 820, 350, 676, 635, 659, 149, 530, 131, 898, 856, 62, 840, 345, 845, 550, 44, 198, 675, 33, 73, 184, 479, 447, 41, 978, 636, 727, 526, 997, 281, 682, 653, 707, 534, 7, 30, 212, 746, 444, 200, 79, 745, 139, 863, 134, 106, 71, 922, 255, 359, 196, 980, 543, 993, 844, 177, 331, 414, 65, 253, 920, 203, 884, 442, 607, 312, 861, 712, 6, 86, 857, 85, 580, 181, 51, 329, 306, 229, 61, 325, 10, 293, 168, 11, 815, 741, 768, 30, 193, 938, 58, 415, 362, 220, 931, 425, 189, 328, 385, 684, 39, 209, 781, 1, 87, 304, 0, 34, 227, 680, 683, 72, 508, 562, 582, 251, 391, 729, 690, 756, 283, 872, 722, 233, 992, 204, 53, 889, 506, 393, 137, 851, 865, 885, 57, 618, 4, 604, 3, 94, 599, 547, 797, 481, 66, 935, 520, 685, 122, 288, 835, 795, 563, 263, 528, 658, 459, 818, 170, 366, 967, 747, 578, 825, 862, 205, 257, 688, 807, 119, 904, 643, 740, 695, 521, 787, 237, 301, 986, 462, 960, 744, 773, 687, 973, 706, 437, 464, 899, 24, 605, 511, 796, 356, 777, 673, 86, 824, 642, 623, 264, 806, 568, 215, 299, 448, 625, 876, 988, 353, 468, 42, 338, 48, 878, 488, 20, 180, 59, 567, 272, 310, 340, 52, 941, 871, 23, 944, 950, 484, 854, 591, 25, 398, 633, 265, 535, 577, 90, 2, 143, 972, 661, 597, 216, 46, 104, 702, 369, 446, 60, 671, 933, 570, 999, 16, 365, 429, 192, 161, 595, 242, 858, 183, 593, 419, 509, 529, 503, 84, 111, 970, 532, 155, 259, 291, 407, 240, 890, 279, 112, 738, 897, 236, 307, 725, 875, 432, 924, 750, 491, 996, 270, 146, 54, 422, 907, 934, 720, 386, 776, 928, 826, 225, 258, 1, 75, 2, 678, 855, 160, 463, 846, 945, 452, 571, 132, 504, 382, 811, 794, 321, 838, 249, 790, 232, 611, 234, 617, 354, 609, 714, 113, 698, 963, 654, 596, 498, 943, 629, 109, 895, 247, 539, 219, 330, 126, 159, 536, 585, 289, 144, 476, 662, 923, 480, 647, 324, 128, 402, 989, 290, 581, 502, 405, 592, 172, 384, 552, 375, 813, 71, 5, 809, 347, 438, 213, 434, 645, 906, 627, 917, 100, 147, 40, 406, 376, 454, 569, 602, 724, 674, 663, 29, 70, 445, 266, 667, 319, 709, 766, 267, 120, 22, 12, 610, 6, 716, 152, 361, 298, 194, 699, 337, 90, 882, 881, 929, 17, 891, 927, 174, 947, 335, 810, 572, 590, 381, 333, 728, 487, 261, 648, 142, 343, 231, 736, 95, 31, 83, 9, 589, 914, 449, 554, 913, 649, 574, 243, 742, 148, 63, 19, 987, 713, 911, 646, 767, 788, 426, 76, 545, 296, 222, 208, 36, 9, 734, 559, 475, 867, 798, 486, 958, 972, 953, 812, 837, 696, 560, 971, 206, 710, 38, 223, 705, 21, 644, 339, 99, 622, 801, 849, 977, 388, 477, 670, 162, 518, 383, 80, 399, 601, 373, 926, 336, 297, 4, 93, 561, 68, 101, 300, 327, 154, 150, 983, 230, 723, 416, 32, 37, 75, 466, 285, 717, 260, 803, 389, 660, 940, 510, 483, 513, 847, 886, 982, 417, 346, 701, 804, 37, 4, 377, 879, 199, 753, 473, 114, 793, 360, 269, 888, 133, 26, 966, 5, 979, 583, 403, 355, 894, 410, 603, 968, 632, 67, 538, 836, 951, 214, 238, 217, 357, 163, 92, 576, 752, 549, 427, 315, 433, 248, 280, 833, 537, 540, 516, 294, 367, 190, 808, 465, 97, 584, 349, 619, 864, 916, 167, 195, 800, 731, 998, 763, 202, 719, 451, 49, 7, 775, 531, 827, 286, 771, 489, 909, 390, 273, 83, 358, 708, 962, 166, 932, 28, 984, 994, 117, 409, 805, 860, 557, 103, 692, 7, 841, 770, 423, 102, 981, 235, 829, 791, 961, 656, 105, 74, 908, 125, 145, 210, 519, 668, 469, 939, 453, 305, 799, 903, 197, 500, 421, 492, 638, 651, 743, 751, 158, 470, 759, 551, 404, 930, 546, 4, 7, 522, 420, 739, 241, 772, 64, 318, 176, 424, 957, 27, 612, 816, 14, 507, 326, 342, 49, 631, 450, 952, 628, 127, 246, 735, 936, 323, 586, 124, 490, 505, 704, 211, 880, 341, 650, 56, 976, 334, 634, 239, 613, 783, 295, 400, 832, 455, 408, 226, 282, 821, 664, 191, 141, 630, 82, 45, 556, 655, 322, 93, 129, 737, 919, 482, 108, 887, 441, 779, 121, 866, 925, 780, 956, 834, 412, 587, 915, 558, 784, 458, 228, 542, 697, 718, 179, 948, 691, 41, 910, 874, 814, 439, 18, 250, 135, 822, 387, 606, 201, 823, 89, 677, 761, 764, 457, 955, 693, 221, 883, 512, 254, 523, 782, 309, 207, 918, 726, 853, 474, 802, 268, 252, 870, 786, 380, 877, 262, 478, 639, 496, 7, 92, 900, 621, 88, 566, 946, 689, 316, 284, 368, 245, 762, 352, 748, 657, 637, 372, 178, 130, 169, 700, 620, 13, 524, 893, 188, 959, 91, 3, 517, 50, 669, 942, 553, 401, 351, 494, 138, 985, 789]

Рисунок 3.2 – Сортування масиву на 1000 елементів

Тестування алгоритму

3.4.3 Часові характеристики оцінювання

В таблиці 3.2 наведені характеристики оцінювання числа порівнянь і числа перестановок алгоритму сортування бульбашки для масивів різної розмірності, коли масив містить упорядковану послідовність елементів.

Таблиця 3.2 – Характеристики оцінювання алгоритму сортування бульбашки для упорядкованої послідовності елементів у масиві

Розмірність масиву	Число порівнянь	Число перестановок
10	27	0
100	1130	0
1000	21023	0
5000	139833	0
10000	319599	0
20000	699137	0
50000	1947681	0

В таблиці 3.3 наведені характеристики оцінювання числа порівнянь і числа перестановок алгоритму сортування бульбашки для масивів різної розмірності, коли масиви містять зворотно упорядковану послідовність елементів.

Таблиця 3.3 – Характеристики оцінювання алгоритму сортування бульбашки для зворотно упорядкованої послідовності елементів у масиві.

Розмірність масиву	Число порівнянь	Число перестановок
10	27	4
100	1328	110
1000	23021	1512
5000	149831	9154
10000	339597	19018
20000	739135	40730
50000	2047679	110332

У таблиці 3.4 наведені характеристики оцінювання числа порівнянь і числа перестановок алгоритму сортування бульбашки для масивів різної розмірності, масиви містять випадкову послідовність елементів.

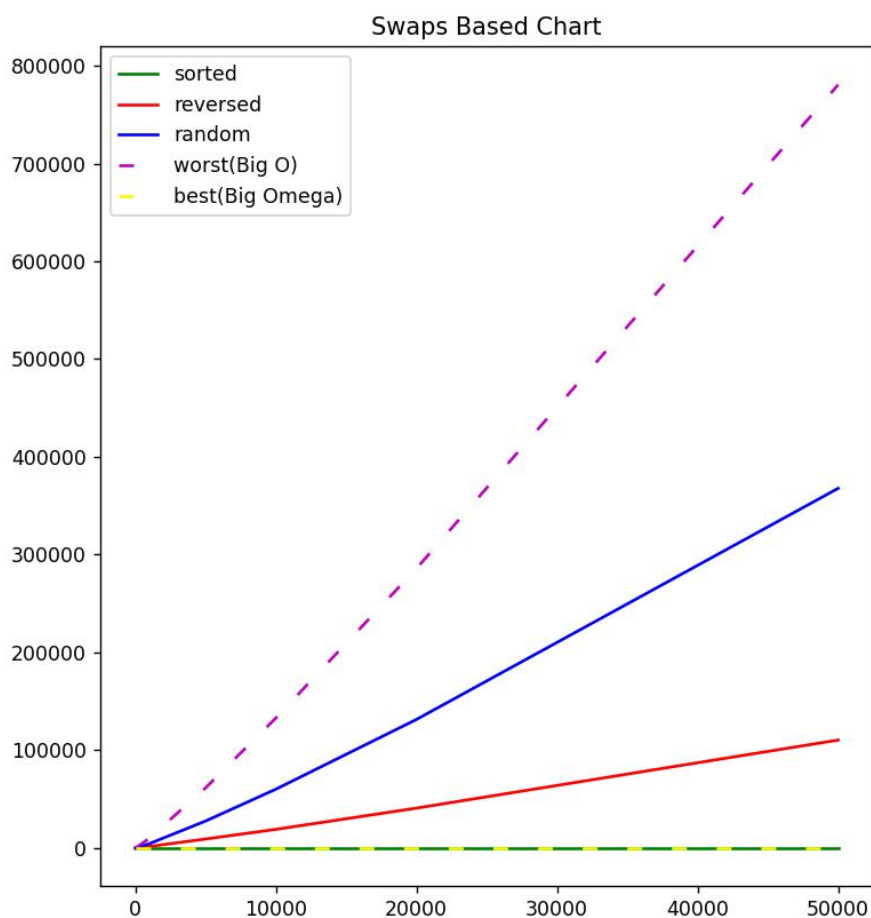
Таблиця 3.4 – Характеристика оцінювання алгоритму сортування бульбашки для випадкової послідовності елементів у масиві.

Розмірність масиву	Число порівнянь	Число перестановок
10	45	9
100	1328	257
1000	24020	4415
5000	154830	27662
10000	359595	60193
20000	759134	131256
50000	2097678	367606

3.4.4 Графіки залежності часових характеристик оцінювання від розмірності масиву

На рисунку 3.3 показані графіки залежності часових характеристик оцінювання від розмірності масиву для випадків, коли масиви містять упорядковану послідовність елементів (зелений графік), коли масиви містять зворотно упорядковану послідовність елементів (червоний графік), коли масиви містять випадкову послідовність елементів (синій графік), також показані асимптотичні оцінки гіршого (фіолетовий графік) і кращого (жовтий графік) випадків для порівняння.

Рисунок 3.3 – Графіки залежності часових характеристик оцінювання



ВИСНОВОК

При виконанні даної лабораторної роботи було вивчено основні методи аналізу обчислювальної складності алгоритмів внутрішнього сортування і оцінити поріг їх ефективності. Було проаналізовано два класичних алгоритми:

- * Бульбашкове сортування

- * Гребінцеве сортування

З'ясовано, що гребінцеве сортування, при великих об'ємах даних, на порядки швидше за бульбашкове:

$$\frac{\text{Обульбашка}}{\text{Огребінець}} = \frac{2.5\text{млрд}}{0.8\text{млн}} = 3.125 * 10^3$$

Також були побудовані графіки залежності кількості перестановок від розміру вхідних даних для обох алгоритмів.

КРИТЕРІЇ ОЦІНЮВАННЯ

У випадку здачі лабораторної роботи до 21.02.2022 включно максимальний бал дорівнює – 5. Після 21.02.2022 – 28.02.2022 максимальний бал дорівнює – 2,5. Після 28.02.2022 робота не приймається

Критерії оцінювання у відсотках від максимального балу:

- аналіз алгоритму на відповідність властивостям – 10%;
- псевдокод алгоритму – 15%;
- аналіз часової складності – 25%;
- програмна реалізація алгоритму – 25%;
- тестування алгоритму – 20%;
- висновок – 5%.