

Київський національний університет імені Тараса Шевченка  
Факультет комп'ютерних наук та кібернетики  
Кафедра інтелектуальних програмних систем  
Алгоритми та складність

Лабораторна робота №2  
«Багатофазне сортування»  
Варіант №1  
Виконав студент 2-го курсу  
Групи ІПС-21  
Міцкевич Костянтин Олександрович

Київ – 2024

**Завдання:** Реалізувати алгоритм багатофазного сортування злиттям.

**Тип даних:** Цілі числа.

**Теорія:**

**Сортування** — Розміщення у визначеному порядку, упорядкування, класифікація чого-небудь.

**Алгоритм сортування** — це алгоритм, що розв'язує задачу сортування, тобто здійснює впорядкування лінійного списку (масиву) елементів.

**Багатофазне сортування** прийнято називати «зовнішнім» сортуванням послідовних файлів, розташованих у зовнішній пам'яті.

**Зовнішнє сортування** — це клас алгоритмів сортування, який може обробляти величезну кількість даних. Зовнішнє сортування потрібне, коли сортовані дані не входять у основну пам'ять обчислювального пристрою (RAM), а замість цього вони повинні знаходитися в повільній зовнішній пам'яті, як правило, на жорсткому диску. Зовнішнє сортування зазвичай використовує гібридну стратегію сортування-злиття. У фазі сортування, шматки даних, достатньо малих для розміщення в основній пам'яті, прочитуються, сортуються та виводяться до тимчасового файлу. У фазі злиття сортовані субфайли об'єднуються в один великий файл

**RAM** — швидкодійна комп'ютерна пам'ять, призначена для запису, зберігання та читання інформації у процесі її обробки.

**Сортування злиттям** — алгоритм сортування, в основі якого лежить принцип «Розділяй та володарюй». В основі цього способу сортування лежить злиття двох упорядкованих ділянок масиву в одну впорядковану ділянку іншого масиву.

Злиття двох упорядкованих послідовностей можна порівняти з перебудовою двох колон солдатів, вишикуваних за зростом, в одну, де вони також розташовуються за зростом. Якщо цим процесом керує офіцер, то він порівнює зріст солдатів, перших у своїх колонах і вказує, якому з них треба ставати останнім у нову колону, а кому залишатися першим у своїй. Так він вчиняє, поки одна з колон не вичерпається — тоді решта іншої колони додається до нової.

Під час сортування в дві допоміжні черги з основної поміщаються перші дві відсортовані підпослідовності, які потім зливаються в одну і результат записується в тимчасову чергу. Потім з основної черги беруться наступні дві відсортовані підпослідовності і так доти, доки основна черга не стане

порожньою. Після цього послідовність з тимчасової черги переміщається в основну чергу. І знову продовжується сортування злиттям двох відсортованих підпослідовностей. Сортування триватиме доти, доки довжина відсортованої підпослідовності не стане рівною довжині самої послідовності.

«Розділяй та володарюй» в інформатиці — важлива парадигма розробки алгоритмів, що полягає в рекурсивному розбитті розв'язуваної задачі на дві або більше підзадачі того ж типу, але меншого розміру, і комбінуванні їх розв'язків для отримання відповіді до вихідного завдання. Розбиття виконуються доти, поки всі підзавдання не стануть елементарними.

### Алгоритм:

Спершу визначимо максимальну кількість записів, яку можна одночасно зберігати в оперативній пам'яті. Для цього створимо масив, розмір якого дорівнюватиме значенню змінної `Size`. На першому етапі зчитуємо `Size` записів, сортуємо їх за допомогою внутрішнього алгоритму сортування, і потім записуємо відсортований набір у файл `INPUT_FILE_A`. Далі зчитуємо наступні `Size` записів, сортуємо їх і записуємо у файл `INPUT_FILE_B`. Цей процес повторюється до тих пір, поки весь вхідний файл не буде розбитий на відсортовані блоки, які по черзі записуються у файли `INPUT_FILE_A` і `INPUT_FILE_B`.

Після того, як весь файл буде розділений на відсортовані блоки, переходимо до етапу їх злиття. Файли `INPUT_FILE_A` та `INPUT_FILE_B` міститимуть відсортовані блоки, однак запис між ними ще не впорядкований, як у випадку сортування злиттям. Ми почнемо процес злиття, подібний до класичного алгоритму, але замість того, щоб записувати дані в масив, ми будемо зберігати їх у новий файл.

Спочатку ми зчитуємо половини перших блоків з кожного з файлів `INPUT_FILE_A` і `INPUT_FILE_B`. Читаємо лише половину блоків, оскільки в пам'яті можемо зберігати одночасно лише `Size` записів, і для злиття нам потрібні дані з обох файлів. Ці частини блоків зливаються в один блок і записуються у файл `TEMP_FILE_A`. Коли одна з половин закінчується, зчитуємо другу частину блоку з того ж файлу. Після завершення злиття одного блоку, решта другого блоку дописується у `TEMP_FILE_A`. Як тільки перші два блоки з `INPUT_FILE_A` та `INPUT_FILE_B` зливаються, наступна пара блоків зливається у файл `TEMP_FILE_B`. Процес продовжується, чергуючи файли `TEMP_FILE_A` і `TEMP_FILE_B` для збереження злитих блоків. Після цього ми отримаємо два файли з відсортованими блоками довжиною  $2 * \text{Size}$ .

Далі процес повторюється: блоки зчитуються з `TEMP_FILE_A` і `TEMP_FILE_B`, а злиті блоки з довжиною  $4 * \text{Size}$  записуються знову у файли `INPUT_FILE_A` та `INPUT_FILE_B`. У результаті, всі блоки зливаються в один відсортований список в одному з файлів.

### Складність алгоритму:

Часова складність алгоритму повинна бути  $O(n\log N)$ , де  $n$  — кількість елементів, які необхідно відсортувати, а  $N$  — кількість списків, що обробляються під час кожної операції злиття. Саме  $O(n\log N)$  виступає як середня складність цього алгоритму.

### Мова реалізації алгоритму: C++

### Модулі програми:

```
//Функція для зчитування даних з файлу у вектор починаючи з заданої позиції.
vector<int> read(const string& filename, int count, int& position){ ... }

//Функція для зчитування одного значення з файлу за вказаною позицією.
int read(const string& filename, int& position){ ... }

//Функція запису у файл.
void write(const string& filename, const vector<int>& values){ ... }

//Функція запису одного значення в файл.
void write(const string& filename, int value){ ... }

//Функція видалення вмісту файлу.
void clear(const string& filename){ ... }

// Функція очищення всіх допоміжних файлів.
void clearAll(){ ... }

//Перевірка чи файл пустий
bool isEmpty(const string& filename){ ... }

//Перевірка чи файл відсортований
bool isSorted(const string& filename){ ... }

//Функція злиття двох відсортованих частин масиву.
void merge(vector<int>& array, int left, int middle, int right){ ... }

//Функція сортування злиттям
void mergeSort(vector<int>& array, int left, int right){ ... }

// Функція розділення вектору на одноелементні вектори
void divide(const string& inputFile, int chunkSize){ ... }

//Функція реалізації багатофазного злиття.
void polyphaseMerge(int initialSize){ ... }

//Основна функція програми для тестування алгоритму.
int main(){ ... }
```

### Інтерфейс користувача:

Вхідні дані (послідовність чисел, розділених пробілами) вводяться через текстовий файл Date File, а результати виводяться у текстовий файл Final Output File.

## Тестові приклади:

### Вхідні дані:

86615 5362 22799 11027 79153 19622 90669 30417 8210 67822 58053 38092 36600 32718 1913 45045 1329 73274 26267 34533 97041 328 50969 14278 36573 24245 70495 97266 26255 31188 98785 50091 21497 34767 62092 1297 9862 95120 70962 65484 32329 26461 22903 66723 66094 61279 28661 52882 22619 74734 10054 96943 3417 96210 3419 10218 30146 2832 51287 4338 12693 81193 80956 3761 21863 81251 71973 61120 89950 86826 79121 63782 35946 17198 2323 38101 51235 72559 74710 27418 52943 95172 97651 25075 32304 6373 32608 94795 90077 75318 64755 1119 81306 89730 75257 65124 54333 42416 30558 28019 14574 66065 11297 79282 54948 67266 11380 54778 41612 30576 30964 74425 82133 79154 1286 44996 50352 80378 72569 2684 25077 36625 29084 2375 3754 38783 26022 82891 93374 49998 97675 98484 49380 50784 2321 71825 86489 72221 25801 47727 40414 35710 12398 17344 3506 80835 64318 77485 52339 26334 43392 19339 14812 57640 13543 86040 63186 39223 36386 536 84545 97485 23727 93290 85422 55552 77810 61950 15033 45322 85662 11281 22268 892 95482 8997 28036 53885 96460 6407 59078 73215 46505 68066 94781 60928 54000 79888 23441 76003 28396 3196 35595 80095 82098 53748 43914 47845 94750 91102 33995 81201 30327 33841 24509 86667 88062 1511 78350 10431 94912 63591 48341 72763 54108 2647 48955 18665 60462 83001 40049 43509 59919 1700 12402 1460 74500 14363 84333 27716 62696 60596 22436 75652 24430 29006 63384 21821 40047 51361 71087 58784 20792 22382 94583 81847 97175 32282 63381 84388 18702 40035 45234 30900 36356 90045 81779 51676 90690 65767 77820 28741 58758 99770 16572 25988 51794 31757 41141 61137 95439 65843 85130 73291 1552 1388 43132 46159 6871 24197 41523 1662 53411 34946 88758 29650 41358 22213 66299 48390 12711 24822 61489 67647 76495 40732 64920 22507 13597 85195 73789 21455 59032 72757 22242 19687 77548 58535 73685 60697 40810 38961 57163 25415 58944 35789 78494 55892 3661 54159 68320 24728 82372 12764 96679 8328 61667 14206 77279 36764 17758 5962 17380 17754 18068 12222 84560 64718 77464 37467 88973 39226 44922 7591 32368 74098 50790 44818 41699 80218 81420 68893 65359 58221 47915 97712 93033 22124 17987 5182 90996 23118 87847 83535 70140 76836 81979 23433 81374 14293 34423 21688 67247 4257 74688 48411 34336 16288 23504 70034 1742 43354 15946 55826 61778 83083 5186 97349 24953 93474 86310 75791 33069 26631 64342 2259 62433 29448 33760 32507 53975 86159 63619 34543 71569 59737 94935 32667 89541 32889 60720 71218 90140 48272 39541 75947 91870 94326 47934 86408 63932 54064 63410 21961 73091 60386 24610 24463 80572 82065 78586 85977 84481 78117 57967 45554 18611 53429 88600 62545 78246 65399 87104 10872 67878 81212 11363 73080 41649 35280 97161 54171 34836 47177 46584 75386 79240 24309 1309 19079 61943 67379 27957 78535 6625 21963 45941 31001 86157 17422 6442 14740 17382 82705 662 55253 23839 48532 77574 89032 93376 77129 7720 51907 29012 25825 45737 583 90595 86216 3011 60139 35438 67427 99116 6413 5212 70128 7014 7559 88384 7095 79186 29927 79760 40362 35474 23024 36307 69116 86210 90544 27776 64510 1777 4272 68695 43739 36612 25283 27514 26467 37743 60854 42557 99666 86667 60109 51073 10193 65149 14919 76190 99179 7249 97053 56165 99978 64666 16046 89024 14282 77118 47581 20529 70220 12975 20113 1549 98850 24588 27062 89590 5460 96473 45965 97521 39232 7460 40645 1411 54128 97351 73741 96895 74729 84593 7147 76119 7634 51999 46853 23434 6775 19987 46569 97014 32913 46586 58907 41360 43870 36694 2270 72309 43966 35253 7807 2192 74348 30596 65213 32861 84359 26359 42326 59290 81259 40469 4051 86429 85240 97040 85525 84561 62448 26884 94350 29648 49473 63625 7327 18896 1618 58097 89233 27379 60409 77813 32619 24999 10144 66652 25179 76724 10583 51415 3398 58809 91698 67108 14875 26215 35697 68064 95485 87358 43511 93454 18547 28627 54617 10956 56589 33390 12677 86584 74842 16169 76864 59077 24700 12610 90389 34583 92523 42460 93544 96908 86551 55192 50430 55984 27976 38970 44409 90836 7033 42256 71048 96954 12862 77004 41927 70458 72484 57994 15567 61391 56811 14459 58009 93581 38850 79236 48496 26176 67633 98799 71581 89186 92742 46943 93682 11886 44291 61495 79935 2864 5776 45059 84132 33552 73813 55628 46631 93318 44189 4598 82105 5959 98699 85063 89528 96166 37728 33120 43 59095 21414 37916 54331 18474 77305 96157 60757 14255 69369 75396 76618 6286 65423 54080 71988 27079 73718 69054 91644 25177 41853 28493 94871 60002 97430 93860 18732 25224 31261 79181 10949 61967 83000 55841 14125 44325 51371 53156 84950 52402 38353 12427 33172 11796 15141 34038 83848 97853 43810 14575 48303 8045 4439 3398 49223 96533 27399 12958 99207 20238 34518 34655 8290 63818 138 21981 58328 32447 29425 23634 76375 20907 22254 58787 53269 82455 96309 22348 6193 11219 41064 83644 42113 31569 77111 57700 7424 79636 30256 31083 84985 28801 21435 51560 81415 57755 27464 55351 83261 73318 53246 11713 83767 99298 63889 504 65479 49245 39507 26253 33319 55890 87037 70727 36339 21159 7330 58078 99381 17655 98249 45373 7234 80064 50665 97485 46054 84087 13716 3364 26953 37054 87529 31416 95760 68831 53210 53388 40842 97256 73285 74105 58705 30601 69495 37513 18696 51211 11495 51097 16849 49682 2167 33072 22713 94701 18672 38021 5039 88025 44626 73817 97056 2755 60757 11576 46635 11137 88679 72756 9994 76470 42883 78974 25310 14563 92374 6859 61599 9395 60917 32357 84152 51390 35152 918 21836 51868 56016 73701 89686 14386 43486 49639 95872 63064 69069 73717 54976 2881 53541 3583 6556 68846 3776 24406 74702 53468 47598 53651 82198 94308 80745 37360 77778 36914 175 8750 11782 49712 31251 83238 27707 77333 93605 7296 43327 37941 66511 88054 18678 57789 95003 72959 534 26899 33462 20447 53510 48491 52155 32162 54221 37137 25258 38846 2924 93949 92333 80895 61984 64298 75586 94946 20710 76972 72103 19787 47009 78858 52157 89997 77923 49073 97713 13072 55490 34163 74041 45817 74470 34815 6756 58927 19054 8665 7686 74957 78106 23184 2225 70837 24378 88734 23910 6602 56807 26583 8579 32409 70695 37183 16840 66809 49232 30032 73003 79002 50532 24829 9464 10446 2763 82255 89620 9842 27750 36832 67961 59213 69237 89729 41698 56182 27513 62687 30908 22341 62062 71465 10380 38882 660 74775 17846 86066 29742 90363 10685 62674 58320 33416 91218 83684 47312 19488 84104 53497 96310 26958 40885 86957 20041 71292 59261 51458 11372 77579 44474 55472 68675 25496 15498 89586 96130 45647 65789 77847 8781 59986 18704 71608 4819 99179 9976 58234 78438 78435 48043 86847 92388 1518 28791 97293 24180 53554 46686 2408 85945 93750 76786 27205 51371 36080 96017 77742 71204 13744 76429 15119 9726 11421 80357 44995 30552 65857 40406 99645 78412 28507 18752 79718 59080 59194 56020 40606 14727 92859 11944 967 53348 73527 42952 58131 98341 13927 26194 52623 76038 6041 66278 1850 17936 93452 35795 71905 8731 14532 68873 15412 19238 18926 44429 80384 22755 44837 52304 82246 31607 24720 10093 82252 91121 81707 68614 91650 84994 89615 6512 82686 3052 63230 69764 87586 28617 14589 28345 67863 56728 7341 4630 30864 22046 1808 66635 88635 31483 97270 56852 36289 49887 88825 17812 56997 68495 56666 15573 13203 67553 20024 82878 70003 23424 85452 17283 46557 24679 77503 64331 91658 72088 78531 35932 58683 84090 32184 58595 17878 67508 77928 95791 43336 31820 73816 29277 15015 78557 54981 76229 73955 55284 36799 49409 69980 16420 22370 30047 11526 6525 92473 85155 44029 31189 12207 65205 82366 8993 21436 21989 94952 66500 42548 64512 73852 19242 29227 44089 90119 53183 91926 19109 60282 81049 64879 33633 5522 21516 63349 6380 57373 45303 96325 65839 53774 66285 36787 70972 80833 81070 55745 74799 44072 46757 13190 58369 91493 68083 43106 25431 90481 91714 68845 57214 82643 83493 42093 57732 65455 62761 40318 31184 74734 88628 86961 60240 19926 26329 93933 18117 15307 69659 6321 86983 10513 47370 46987 52992 33512 83282 30410 85609 95461 5289 72699 7183 80555 56725 50046 53245 71562 6609 76071 62897 68521 40647 16007 25825 35378 31554 53970 14308 89239 472 29841 99694 78138 34928 79536 82400 57814 73325 37129 82867 87003 82867 55590 99463 98709 26706 25620 433 22282 53802 452 24381 13261 77901 55484 68225 49500 89833 29104 49073 88492 67787 37783 91415 57296 47036 87010 63889 70005 88243 72028 40647 41364 38697 35044 3562 28585 50566 14672 67860 50107 38852 97417 75298 18070 75811 46281 16417 62501 91891 88081 17385 81563 23009 21949 19148 77300 79266 91417 85519 87327 53297 17818 85948 24473 50207 8193 69943

## Вихідні дані:

43 138 175 328 433 452 472 504 534 536 583 660 662 791 892 918 967 1119 1286 1297 1309 1329 1388 1411 1460 1511 1518 1549 1552  
1618 1662 1667 1700 1742 1777 1808 1850 1913 2167 2192 2225 2259 2270 2321 2323 2375 2408 2573 2647 2684 2755 2763 2832 2864 2881 2924 3011 3052 3196  
3364 3398 3398 3417 3419 3506 3562 3583 3661 3754 3761 3776 4051 4257 4272 4292 4338 4439 4453 4598 4630 4663 4819 5039 5182 5186 5212 5289 5362 5460  
5522 5525 5776 5959 5962 6041 6193 6286 6321 6373 6380 6407 6413 6442 6512 6525 6556 6602 6609 6625 6756 6775 6859 6871 7014 7033 7095 7147 7183 7234  
7249 7296 7327 7330 7341 7424 7452 7460 7559 7591 7634 7686 7720 7807 8045 8193 8210 8290 8328 8579 8665 8731 8750 8781 8993 8997 9395 9464 9499 9726  
9842 9862 9976 9994 10054 10093 10144 10193 10218 10380 10431 10446 10513 10583 10685 10872 10949 10956 11027 11137 11219 11281 11297 11363 11372  
11380 11421 11495 11526 11576 11713 11782 11796 11886 11944 12049 12207 12222 12367 12398 12402 12427 12610 12677 12693 12711 12764 12862 12958 12975  
13072 13190 13203 13261 13543 13597 13716 13744 13927 14125 14206 14255 14278 14282 14293 14308 14363 14386 14434 14459 14532 14563 14574 14575 14589  
14672 14727 14740 14812 14875 14919 15015 15033 15119 15141 15307 15412 15498 15567 15573 15946 16007 16046 16169 16288 16417 16420 16572 16840 16849  
17037 17198 17283 17344 17380 17382 17385 17422 17655 17754 17758 17812 17818 17846 17878 17936 17987 18068 18070 18117 18474 18547 18572 18611 18665  
18672 18678 18696 18702 18704 18732 18752 18896 18926 19054 19079 19109 19146 19148 19238 19242 19248 19339 19413 19488 19622 19687 19787 19926 19987  
20024 20041 20113 20238 20447 20529 20710 20792 20907 21159 21414 21435 21436 21455 21497 21516 21527 21688 21794 21821 21836 21863 21949 21961 21963  
21981 21989 22046 22124 22213 22242 22254 22268 22282 22341 22348 22370 22382 22436 22507 22619 22713 22755 22799 22903 23009 23024 23118 23184 23331  
23424 23433 23434 23441 23504 23634 23727 23839 23910 24180 24197 24245 24309 24378 24381 24406 24430 24463 24473 24509 24564 24588 24610 24679 24700  
24720 24728 24791 24822 24829 24953 24999 25075 25077 25177 25179 25224 25258 25283 25310 25336 25415 25431 25496 25620 25801 25825 25825 25988 26022  
26176 26194 26215 26253 26255 26267 26329 26334 26359 26461 26467 26583 26631 26706 26884 26899 26953 26958 27062 27079 27205 27379 27399 27418 27464  
27513 27514 27707 27716 27750 27776 27957 27976 28019 28036 28345 28396 28493 28507 28585 28617 28627 28661 28741 28791 28801 29006 29012 29084 29104  
29227 29277 29425 29448 29648 29650 29742 29841 29927 30032 30047 30146 30256 30327 30410 30417 30451 30552 30558 30576 30596 30601 30864 30900 30908  
30964 31001 31083 31184 31188 31189 31251 31261 31416 31483 31547 31554 31569 31607 31757 31820 32162 32184 32282 32304 32329 32357 32368 32409 32447  
32507 32608 32619 32667 32718 32756 32861 32889 32913 33069 33072 33120 33172 33319 33390 33416 33462 33512 33552 33633 33760 33841 33995 34038 34132  
34163 34336 34423 34518 34533 34541 34543 34583 34655 34767 34815 34836 34923 34928 34946 35044 35152 35253 35280 35378 35438 35474 35595 35697 35710  
35789 35795 35932 35946 36075 36080 36289 36307 36339 36356 36386 36422 36573 36625 36692 36764 36787 36799 36832 36914 37054 37129 37137  
37183 37360 37459 37467 37496 37513 37658 37728 37743 37783 37916 37941 38021 38092 38101 38353 38561 38697 38783 38846 38850 38852 38882 38961 38970  
39223 39226 39232 39408 39507 39541 40035 40047 40049 40318 40360 40362 40406 40414 40469 40606 40608 40645 40647 40647 40732 40810 40842 40881 40885  
41064 41141 41214 41358 41360 41364 41523 41612 41649 41698 41699 41802 41853 41927 42050 42093 42113 42256 42326 42359 42416 42460 42548 42557 42818  
42883 42952 42953 43076 43106 43132 43327 43336 43354 43392 43486 43509 43511 43739 43810 43870 43914 43966 44029 44072 44089 44189 44291 44325 44409  
44429 44474 44626 44818 44837 44922 44995 44996 45045 45059 45234 45303 45322 45373 45554 45647 45737 45817 45941 45965 46054 46159 46281 46505 46557  
46569 46584 46586 46631 46635 46686 46757 46853 46943 46987 47009 47036 47153 47177 47312 47370 47581 47598 47727 47845 47915 47934 48043 48272 48303  
48341 48390 48411 48482 48491 48496 48532 48955 49073 49073 49223 49232 49245 49380 49409 49473 49500 49639 49682 49712 49887 49998 50046 50091 50107  
50207 50352 50430 50532 50566 50665 50784 50790 50969 51073 51097 51210 51211 51235 51287 51361 51371 51371 51390 51415 51458 51560 51676 51794 51868  
51907 51999 52155 52157 52159 52271 52304 52339 52402 52548 52623 52882 52920 52943 52992 53156 53183 53210 53245 53246 53269 53297 53348 53388 53403  
53411 53429 53468 53497 53510 53541 53554 53651 53748 53774 53802 53885 53970 53975 54000 54064 54080 54108 54128 54159 54171 54221 54331 54333 54377  
54617 54778 54948 54976 54981 55192 55214 55253 55284 55351 55472 55484 55490 55552 55590 55628 55745 55826 55841 55890 55892 55984 56016 56020 56165  
56182 56337 56589 56666 56716 56725 56728 56792 56807 56811 56852 56997 57163 57214 57296 57373 57640 57700 57732 57755 57789 57814 57967 57994 58009  
58053 58075 58078 58097 58131 58221 58234 58320 58328 58369 58535 58595 58683 58705 58758 58784 58787 58809 58907 58910 58927 58944 59032 59077 59078  
59080 59095 59194 59213 59261 59290 59737 59919 59986 60002 60109 60139 60240 60282 60386 60409 60462 60596 60697 60720 60757 60757 60854 60917 60928  
61120 61137 61279 61391 61489 61495 61599 61667 61778 61943 61950 61967 61984 62062 62092 62433 62448 62501 62545 62674 62687 62696 62761 62897 63026  
63064 63186 63230 63349 63381 63384 63392 63410 63497 63591 63619 63625 63782 63818 63889 63889 63932 64298 64318 64331 64342 64480 64510 64512 64666  
64718 64755 64879 64920 65124 65149 65205 65213 65309 65359 65399 65423 65455 65479 65484 65767 65789 65839 65843 65857 66065 66094 66224 66278 66285  
66299 66500 66511 66594 66635 66652 66723 66809 66842 67108 67247 67266 67379 67427 67493 67508 67549 67553 67616 67633 67647 67787 67822 67860 67863  
67878 67961 68064 68066 68083 68225 68320 68495 68521 68614 68675 68695 68826 68831 68845 68846 68873 68893 69054 69069 69116 69237 69369 69495 69659  
69764 69943 69980 70003 70005 70034 70128 70140 70220 70458 70495 70695 70727 70837 70962 70972 71048 71087 71204 71218 71225 71276 71292 71465 71562  
71569 71581 71608 71825 71905 71973 71988 72028 72088 72103 72119 72221 72309 72484 72559 72569 72699 72756 72757 72763 72959 73003 73080 73091 73215  
73274 73285 73291 73318 73325 73478 73527 73685 73701 73717 73718 73741 73813 73816 73817 73852 73955 74041 74098 74105 74348 74425 74470 74500 74688  
74692 74702 74710 74729 74734 74734 74775 74789 74799 74842 74957 75257 75298 75318 75386 75396 75586 75652 75791 75811 75947 76003 76038 76071 76119  
76190 76229 76375 76429 76470 76495 76618 76724 76786 76836 76864 76970 76972 77004 77111 77118 77129 77279 77300 77305 77305 77333 77464 77485 77503  
77514 77548 77574 77579 77742 77778 77810 77813 77820 77847 77901 77923 77928 78070 78106 78117 78138 78246 78350 78412 78435 78438 78494 78531 78535  
78557 78586 78858 78974 79002 79121 79153 79154 79181 79186 79236 79240 79265 79266 79270 79282 79536 79612 79636 79718 79760 79888 79891 79935 80064  
80095 80218 80357 80378 80384 80555 80572 80745 80833 80835 80895 80956 81049 81070 81193 81201 81212 81251 81259 81306 81374 81415 81420 81563 81707  
81779 81847 81979 82065 82098 82105 82133 82198 82246 82252 82255 82366 82372 82400 82455 82643 82686 82705 82867 82867 82878 82891 83000 83001 83083  
83238 83261 83282 83412 83464 83493 83535 83644 83684 83767 83848 83920 84087 84090 84101 84104 84125 84132 84152 84333 84359 84388 84481 84545 84560  
84561 84593 84777 84950 84985 84994 85063 85130 85155 85195 85240 85422 85452 85519 85525 85609 85662 85945 85948 85977 86040 86066 86157 86159 86210  
86216 86310 86362 86369 86408 86429 86489 86551 86584 86615 86667 86667 86826 86847 86957 86961 86983 87003 87010 87037 87104 87327 87358 87529 87586  
87847 88025 88054 88062 88081 88243 88384 88492 88600 88628 88635 88679 88734 88758 88825 88973 89024 89032 89186 89233 89239 89528 89541 89586 89590  
89615 89618 89620 89686 89729 89730 89833 89950 89997 90045 90077 90119 90122 90140 90363 90389 90481 90544 90552 90595 90669 90690 90836 90996 91102  
91121 91218 91415 91417 91493 91644 91650 91658 91698 91714 91870 91891 91926 92333 92374 92388 92473 92523 92742 92859 92998 93033 93290 93318 93374  
93376 93452 93454 93474 93544 93581 93605 93658 93682 93750 93860 93933 93949 94308 94326 94350 94583 94701 94750 94781 94795 94871 94912 94935 94946  
94952 95003 95120 95172 95439 95461 95482 95485 95760 95791 95872 96017 96130 96154 96157 96166 96210 96309 96310 96325 96460 96473 96533 96679 96895  
96908 96943 96954 97014 97040 97041 97053 97056 97161 97175 97256 97266 97270 97293 97349 97351 97417 97430 97485 97485 97521 97651 97675 97712 97713  
97853 98249 98341 98378 98484 98699 98709 98785 98799 98850 99116 99179 99179 99207 99298 99381 99463 99645 99666 99694 99770 99978

### **Висновок:**

Багатофазне сортування злиттям застосовується, коли необхідно відсортувати великі обсяги даних, які не поміщаються в оперативну пам'ять. Цей метод дозволяє оптимізувати процес сортування, зменшуючи витрати часу, навіть за умови роботи з дуже великими файлами. Алгоритм ефективно розподіляє обчислювальні ресурси, забезпечуючи сортування даних частинами з їх поступовим злиттям у впорядковану послідовність, що дозволяє уникнути надмірних витрат пам'яті й часу. Завдяки цьому, багатофазне сортування є корисним у випадках, коли критично важливим є збереження балансу між часом виконання й ресурсами системи.

### **Використані літературні джерела:**

- Лекція 3
- [Багатофазне сортування - реферати та учбові матеріали на um.co.ua](#)
- [Сортування злиттям — Вікіпедія \(wikipedia.org\)](#)
- [Polyphase merge sort - Wikipedia](#)