**Звіт до**

**лабораторної роботи №1**

**З АСД**

Павлюченко Артем

ІПЗ-11/1

# **Умови завдання:**

Написати програму мовою C# з можливістю вибору різних алгоритмів пошуку. Продемонструвати роботу (ефективність, час виконання) програм на різних структурах даних (масив, лінійний зв’язаний список), з різними умовами, що забезпечують зменшення часу виконання. Навести аналіз отриманих результатів. Реалізувати алгоритми:

* пошуку перебором елемента масиву, що дорівнює заданому значенню.
* пошуку з бар'єром елемента масиву, що дорівнює заданому значенню.
* бінарного пошуку елемента масиву рівного заданому значенню.
* бінарного пошуку елемента масиву, рівного заданому значенню, в якій нове значення індексу m визначалося б не як середнє значення між L і R, а згідно з правилом золотого перерізу.

# **Аналіз задачі:**

Усі алгоритми працюватимуть з цілими числами, аби краще проаналізувати результати. Для усіх видів пошуку із масивом ми використовуємо лише один клас Program, у якому масив створюється та виконуються алгоритми над ним, а вже для зв’язного списку мені довелося створити додатковий клас Node, у якому я і реалізував створення цього списку, вивід його на екран, заповнення списку, та його ініціалізацію. Та усі алгоритми пошуків у зв’язному списку ми виконуємо у класі Program.

Робота з масивом: за допомогою оператора switch користувач обирає роботу з масивом, створює його, та за допомогою новоствореного меню вибирає ті дії і алгоритми, які хоче провести над масивом.

Робота зі зв’язним списком: за допомогою оператора switch користувач обирає роботу зі зв’язним списком, створює його, та за допомогою новоствореного меню вибирає ті дії і алгоритми, які хоче провести над списком. (у обох випадках меню не відрізнятиметься)

Для повноцінної перевірки працездатності моїх алгоритмів я перевіряв один і той самий масив або список на пошук багатьох елементів, різними кроками, різними послідовностями дій, та отримав досить інформативні результати.

# **Структура основних вхідних та вихідних даних:**

Під час виконання кожного алгоритму пошуку вхідними даними буде виступати масив цілих чисел та елемент, який ми шукаємо. Вихідними даними виступатиме позиція шуканого елемента.

# **Алгоритм розв'язання задачі:**

**Лінійний пошук:**

Для масиву:

int x = Convert.ToInt32(Console.ReadLine());

int i = 0;

bool flag1 = false;

while ((i < kolvo) && (!flag1))

{

if (numbers[i] == x)

{

flag1 = true;

Console.WriteLine("Index of your element " + i);

}

i++;

}

if (i == kolvo) Console.WriteLine("Entered number don`t exist in array");

Для списку:

int key = Convert.ToInt32(Console.ReadLine());

int x = 0;

bool flag2 = false;

while ((x < kolvo1) && (!flag2))

{

if (head.data == key)

{

flag2 = true;

Console.WriteLine("Index of your element " + x);

}

head = head.next;

x++;

}

if (x == kolvo1) Console.WriteLine("Entered number don`t exist in array");

**Пошук з бар’єром:**

Для масиву:

int x = Convert.ToInt32(Console.ReadLine());

int k = 0;

h1numbers[kolvo] = x;

while (h1numbers[k] != x)

{

k++;

}

if (k == kolvo) Console.WriteLine("Element hasn`t been found");

else Console.WriteLine("Index of your element " + k);

Для списку:

int x = Convert.ToInt32(Console.ReadLine());

int k = 0;

head.AddToEnd(x);

while (head.data != x)

{

head = head.next;

k++;

}

if (k == kolvo1) Console.WriteLine("Entered number don`t exist in array");

else Console.WriteLine("Index of your element " + k);

**Бінарний пошук (з або без правила золотого перерізу):**

Для масиву:

int l = 0;

int r = kolvo - 1;

int m = 0;

int x;

Console.WriteLine("Enter x");

x = Convert.ToInt32(Console.ReadLine());

do

{

m = r - ((r - l) / 2);

if (hnumbers[m] < x) l = m + 1;

else r = m - 1;

} while ((l <= r) && (hnumbers[m] != x));

if (hnumbers[m] == x)

{

Console.WriteLine("X in array:");

Console.WriteLine("a[" + m + "]= " + hnumbers[m]);

}

# else Console.WriteLine("X isn't included in array");

Для списку:

int l = 0;

int r = kolvo1;

int m = 0;

int x;

Sort(head);

Console.WriteLine("Your linked list after sorting: ");

head.Print();

Console.WriteLine();

Console.WriteLine("Enter x");

x = Convert.ToInt32(Console.ReadLine());

do

{

m = r - ((r - l) / 2);

if (GetElement(head,m) < x) l = m + 1;

else r = m - 1;

} while ((l <= r) && (GetElement(head, m) != x));

if (GetElement(head, m) == x)

{

Console.WriteLine("X in linked list:");

Console.WriteLine("Index of your element " + (m-1));

}

else Console.WriteLine("X isn't included in array");

Допоміжні алгоритми: Сортування списку і отримання елементу зі списку:

static void Sort(Node obj)

{

int tmp;

Node index = null;

while (obj != null)

{

index = obj.next;

while (index != null)

{

if (obj.data > index.data)

{

tmp = obj.data;

obj.data = index.data;

index.data = tmp;

}

index = index.next;

}

obj = obj.next;

}

}

private static int GetElement(Node start, int position)

{

for (int i = 0; i < position-1; i++)

{

start = start.next;

}

return start.data;

}

# **Код програми:**

Увесь код програми знаходиться в репозиторії GitHub:

https://github.com/Pavich123/Programming-Algorithms

# **Набір тестів:**

1. Елемент 728, знаходиться на 38 позиції:

Масив на 100 елементів: 420 841 655 171 903 957 687 298 167 966 895 591 195 895 415 659 706 740 230 860 40 909 850 394 384 273 791 524 169 172 953 934 291 276 806 786 712 339 728 431 718 731 393 743 943 540 573 175 350 427 201 732 237 43 489 511 990 262 787 631 167 164 130 994 13 961 301 919 90 628 947 368 12 799 142 308 516 107 450 844 699 617 174 741 971 222 698 248 786 295 795 450 551 98 264 555 602 398 731 982

1. Елемент 4142, знаходиться на 956 позиції:

Масив на 1000 елементів: 4918 6110 3168 1415 7111 5374 887 2939 4235 277 3353 7737 8572 5170 8905 645 7038 7959 6496 8995 662 2081 3703 46 1392 5587 1189 2199 5228 5560 2959 705 7777 2633 6549 5831 7755 3674 711 2035 9253 9975 3556 7477 3982 1435 6553 609 1296 3213 2078 5651 3143 2431 8077 2837 2407 3122 23 1525 4185 8687 7711 8674 7317 2649 9960 5939 8621 3075 2889 3364 7249 4462 9741 686 8524 6225 6064 9957 9033 580 904 2015 3483 7307 7561 5347 4555 3712 3424 4633 3651 9186 7849 566 2265 4881 160 1334 1475 614 1988 8221 324 8713 8402 8681 2689 7391 4313 6181 7058 66 2491 3605 7784 5696 5192 11 5087 4614 1384 4910 9650 8256 9712 8063 6612 9175 8421 3643 6066 4731 8482 8419 8592 2682 1691 4769 8904 8880 2658 7164 9399 7242 7575 3585 6695 4244 2782 6568 9689 150 6246 6861 9230 7078 8571 2068 9000 340 2069 3513 8970 670 116 2327 1584 4071 5013 5103 4005 423 1106 6206 1956 4220 5510 2409 682 6127 1368 2369 6394 1853 3954 5916 8484 1622 9189 1514 4110 9623 5768 8564 6812 9144 8193 8730 7126 5248 2002 2625 9230 7678 2563 9266 9044 40 4905 5010 1568 6163 1387 2873 8972 9700 7118 7117 6715 4200 3843 9961 4881 3500 993 4382 4654 2541 9394 2811 6026 6780 5282 5434 4126 8743 3138 8716 9289 4687 6872 8445 6717 4179 9945 7947 8236 2895 9592 7111 2027 1076 2015 2926 1406 2041 7743 5731 6686 8181 4613 6503 646 2095 8984 4788 881 5953 8747 229 6563 8402 7828 2028 7328 5397 3244 702 3555 3045 6145 1759 2948 2283 785 4950 4765 2356 4029 2085 1001 7406 2031 1109 75 369 7800 4623 5195 5158 7066 2284 4148 9363 549 3624 3248 9986 5598 6008 8796 7041 2177 3641 2036 2854 3555 8362 1310 4034 23 8525 1925 6663 9228 9156 6372 6719 1954 6960 7597 8621 5507 8396 5978 3862 7611 3585 1735 7160 1702 4779 6757 8020 3288 3959 5230 8390 9073 7220 6813 9437 3313 1162 5135 8541 360 7484 135 1393 7252 6529 8033 8638 8410 176 1534 3780 7662 8174 5243 9970 6628 4149 2332 5244 1768 3904 3375 5269 3926 7982 7646 4733 147 8159 5308 4346 3262 7437 3502 127 3450 342 9908 5173 6745 8119 9610 3113 2425 1451 728 900 1978 6842 2810 9164 8829 9891 6773 6456 4110 4865 7467 9269 8882 3300 8491 251 4867 7188 519 225 4672 5116 6521 6286 4240 7158 8498 3649 4294 262 2844 2476 7254 6747 2755 3304 5350 6144 5516 3370 664 7045 6017 8584 2875 640 6291 3446 9627 9359 8246 5237 933 504 6227 6862 322 6524 4924 1671 1394 3125 2163 3848 2021 4991 2016 2136 546 5188 4901 8722 1672 7148 9561 7627 9098 7936 3412 3601 868 5053 4021 4934 2016 7607 1543 6751 520 5893 2983 8826 1221 3846 1977 7538 4883 2170 6563 7883 8624 4155 2900 4440 4458 9524 3565 3784 943 8600 7763 2385 3113 1324 803 6340 9104 7228 1832 4414 3449 5264 1616 4653 2206 6075 7502 7826 5172 2023 2745 6929 1374 5528 4977 6713 2153 9581 477 2492 4043 7759 5809 1920 8129 598 5714 9897 3044 5636 8434 7654 338 2150 4435 3360 2539 8247 2234 8383 2022 5739 8612 8919 5855 834 1012 7584 6346 4089 4188 9523 6752 9340 372 5689 9455 9696 6523 1608 362 7605 4783 9535 3589 5090 6592 9224 1093 1618 4174 3905 7347 2094 470 8304 9147 6889 6064 7295 9586 8129 3551 8954 1448 8910 902 997 1779 8745 3904 2843 1725 626 8021 4417 957 9076 5331 766 4242 1787 6491 4729 9915 284 2177 4659 8871 2068 6542 2567 3632 9227 2022 2232 4054 5828 8088 4678 4188 5595 7446 2347 7713 1332 2181 6721 4072 6052 7348 72 1558 5299 3054 7799 1638 8821 9039 1165 6734 6243 2126 9711 2204 1338 9211 2497 8604 5789 364 5128 988 653 6577 8646 4341 4144 7016 8583 8103 5455 588 2819 4720 6471 1009 8332 6174 4223 595 5232 6789 6923 7943 7945 3470 7735 144 6376 2120 9683 8117 8282 5689 2220 9083 906 8722 4407 3458 7493 1806 456 3062 1280 5656 9306 4992 5733 329 880 6323 4381 5195 5131 8338 4065 2710 8632 5177 6606 4001 640 6464 8419 7338 2305 7130 2500 7388 104 9609 1768 765 3101 3427 6334 3861 6664 2289 4163 2744 4411 6047 1241 3361 3736 3087 558 3882 5018 8195 91 9229 6852 3493 1166 3992 4642 3941 3351 2176 2492 8345 226 1270 4555 3617 2095 1705 2004 204 6045 6343 1015 3863 9590 4593 5223 5059 3603 9218 6573 8617 2370 1908 9519 2538 3912 9608 2261 2343 9219 2723 8938 1987 252 6066 5821 9971 8805 119 993 8853 1879 4815 2151 3748 8215 2990 3903 6573 8768 9583 339 4133 5603 3875 5975 8318 1751 2018 9704 2487 9443 9660 985 3322 7404 9027 3612 3524 8771 5252 6254 3484 8225 7719 6738 7555 9757 5772 4322 923 5705 5687 3575 9636 2385 4805 6384 6376 91 7503 8221 6787 415 8506 9409 2218 3830 8828 6343 9188 9378 379 7802 3517 3329 6855 5908 7883 7136 8420 8560 5979 7695 8781 6781 3757 6086 1349 938 2600 2643 7235 3434 1268 7031 9466 3069 9719 8310 4521 3725 929 9428 5134 1545 5327 7885 58 6307 5529 8897 8500 9239 1671 8942 2243 9092 1633 1726 5653 6132 2482 7890 3744 6545 2143 6945 6534 6486 3863 3787 6189 9573 2616 4695 7632 6550 2561 7237 1454 5871 6028 5042 5409 3702 4142 7995 1764 2325 4789 375 1436 7993 2657 8388 1244 3039 5685 8588 9402 8382 1351 3572 2445 1743 2709 8927 6624 6975 1311 5692 6531 4396 272 9781 105 7440 2482 43 2403 4147 5181 4209 1698 3489 2351 8195 6916 4227

1. Елемент, відсутній у масиві:

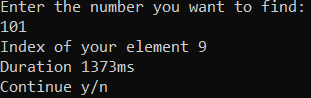
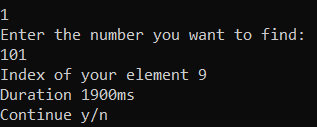
Масив на 1000 елементів: 7088 4315 295 2524 4872 9859 8651 5004 1942 9074 382 8463 3429 7188 4908 9021 5124 6359 4351 8266 677 4736 7900 3824 4137 863 17 1100 8975 7571 7001 3202 6568 3205 8516 3010 6387 1107 1977 8471 5483 1918 4969 5993 5211 9761 5801 6804 1364 9867 5619 6605 7035 8999 3783 2353 6415 6471 8386 4010 9842 7551 6028 4370 2074 7180 1896 225 8671 1898 2635 4017 4383 5879 2784 8759 9767 1908 8613 4376 5062 3213 9736 9108 1952 397 6166 7569 9421 6164 6595 9916 2721 7966 8629 7931 5889 599 3919 8030 7866 5577 8132 9465 7233 1603 2222 1157 6215 5024 2586 4507 7857 4011 8948 6629 7815 6919 2418 1678 1014 4327 803 2508 5303 2718 1297 6417 7250 4853 2870 9168 7988 583 6510 9485 5081 272 1876 350 8174 5010 1354 4398 3579 2088 2059 8709 9018 2000 117 8969 8180 2242 7017 3539 4774 5625 4163 4515 307 5805 3906 1363 2155 3417 6519 7275 7501 9462 1549 7544 5044 2069 3503 6003 2974 6405 8928 3215 659 2587 7398 5251 4736 3900 989 5746 3566 2971 4711 9455 6108 7360 44 2370 1105 9990 2244 163 5568 4784 1208 9555 452 2572 3926 6111 8739 1014 566 8369 6696 948 3856 7719 8406 8654 6626 8254 2429 773 3710 4531 4752 2093 1436 7683 2026 1133 4899 2984 4161 8766 7646 5875 1379 7842 4800 2165 9973 4877 7007 2552 2406 6342 2760 5160 3504 2324 3964 2450 3364 3989 7733 4796 1074 6677 4804 8358 1137 6243 4086 7606 6114 7582 4208 7930 3301 7981 6340 564 3855 4462 8280 7551 3765 1158 2126 8409 9333 6275 4179 9701 7169 2449 9619 173 1033 2851 4802 4702 3039 6441 1028 3731 792 9401 6437 4823 2135 4829 1859 5523 5984 3400 8595 8901 5708 183 1031 9915 4552 6395 9025 4861 2064 4384 437 3666 7962 4036 6898 451 3180 1639 7525 7413 3434 4549 6760 4364 4720 7302 6275 4504 4417 8655 3717 3770 3853 719 4464 851 1821 4886 150 6643 7416 6167 1668 6407 8964 2772 6861 8099 7931 1409 2344 4346 5874 1183 5467 1160 3423 6666 5196 7250 120 4521 445 3408 667 6667 9812 7244 9571 6048 8630 8294 1489 882 9996 7267 2882 353 5399 1949 442 8176 6572 3008 6312 9371 7895 2671 5251 3305 7427 5154 9689 2899 6524 2895 5723 8259 5741 2297 2960 9616 8527 1884 2779 4050 2857 4993 1186 8200 8277 3071 1268 3248 6809 1944 7949 7436 7096 1296 8772 7141 1994 6267 8620 3476 8604 8589 4358 7102 1544 4622 4611 3103 8988 825 9649 4689 230 2263 6514 2903 1486 7050 5027 4357 3887 6442 6090 4581 4946 8286 1164 4445 3525 5818 7623 2261 3264 9302 5446 4267 635 4084 6657 3655 8459 8165 4259 5984 2295 3260 7207 4833 4782 5869 5656 4943 1240 4263 9589 2162 2500 9777 2156 3257 3458 167 9577 3171 3202 7389 1425 927 6817 2247 2268 7401 394 1373 5898 5722 2184 106 2287 1686 1080 6331 9663 7655 163 2680 1021 9001 9712 3284 1768 858 1928 3400 197 8293 8587 1089 2783 4905 1836 6280 8016 2535 3602 8254 4550 9867 8364 3868 9977 2395 7491 470 2178 7127 503 1922 3008 522 6368 2424 1215 3533 480 1411 5474 6993 1176 7605 7134 1096 7322 7381 9850 4799 8314 7128 4054 1909 8130 1154 637 5844 3306 9373 3367 1459 1222 3070 7790 6665 8080 2262 8537 9411 5066 4483 2056 2192 2781 7556 8692 760 6734 8882 5072 110 620 7379 8813 3374 7868 1100 2392 5214 1788 5370 228 1107 8043 4016 5772 8105 9815 469 3015 5061 8843 439 9732 3831 5072 1863 9127 574 2462 9877 9110 4423 4301 3257 839 3843 4256 4416 8797 6981 9870 3323 7624 9696 4256 950 4148 8765 1785 587 944 6265 5868 12 1267 182 7646 4982 8301 6006 1972 1818 2753 1911 6259 5804 6805 4787 3177 1929 3850 5399 1672 6034 5190 5520 2814 37 9575 4123 7715 363 8789 1875 8933 2846 8555 4289 1991 658 6197 9274 6114 2792 5009 8053 571 5714 3437 8452 4145 9361 5588 9856 6737 5545 4594 9833 4821 5746 7367 7609 5407 4179 8291 1610 3028 878 2977 3414 7248 2516 2797 2520 5731 4575 9284 8880 1026 7137 4950 7100 6599 1123 9978 8354 4775 8933 5138 3389 8252 8721 9467 6244 3291 8588 3868 1936 4501 3400 5025 9692 2737 24 1204 1629 6565 3069 4125 2162 6260 5713 8808 7684 797 267 1010 4284 4201 9937 6834 4095 5740 9588 5162 8526 3049 6552 9229 7143 708 7349 4380 7626 2113 5258 4363 6576 9919 8349 1790 1706 4808 2977 7128 2539 9913 1784 5448 3024 7578 9583 7734 4564 6565 930 3953 3148 4861 2677 8580 13 3840 6982 1455 8910 1334 1183 5572 5538 5904 4433 4365 5851 8147 5129 9287 2763 2461 2624 8613 1265 1104 6205 9564 1124 9615 9815 1061 1183 1306 1216 1715 7242 9769 1777 7866 7825 1522 8218 1205 8730 6212 9909 7120 3145 5219 1884 6416 1436 1643 1190 688 2238 4064 7315 8908 7634 7417 331 9295 1519 122 4389 4233 4689 2719 7122 6081 6371 7262 1462 1242 4243 1420 9882 5053 1195 9085 6420 5905 7731 3400 9624 9539 116 528 9477 3179 2455 2868 232 409 1192 8922 9686 8609 1824 5677 2432 427 8096 5802 45 4173 182 9948 6445 819 4181 2263 7714 8549 998 4426 1564 8118 497 4849 4117 4161 3242 3943 3627 3503 7031 1053 51 5320 1734 1274 3230 5517 6654 5994 7808 1930 3356 5451 9358 168 4083 8658 8998 192 5154 1683 9411 6765 7358 1568 8112 6974 1560 8270 7184 4154 2176 6542 7142 9128 9897 1125 9085 2908 9033 2197 1895 5003 6618 9634 4763 5045 5490 3949 79 4583 4945 3435 8349 5348 1642 3285 7962 166 3161 6255 3958

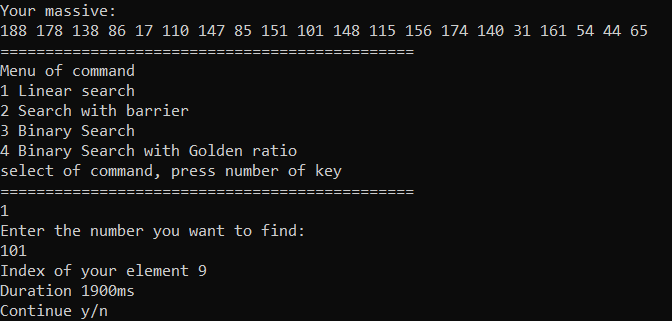
1. Елемент 39709 , знаходиться на 9751 позиції:

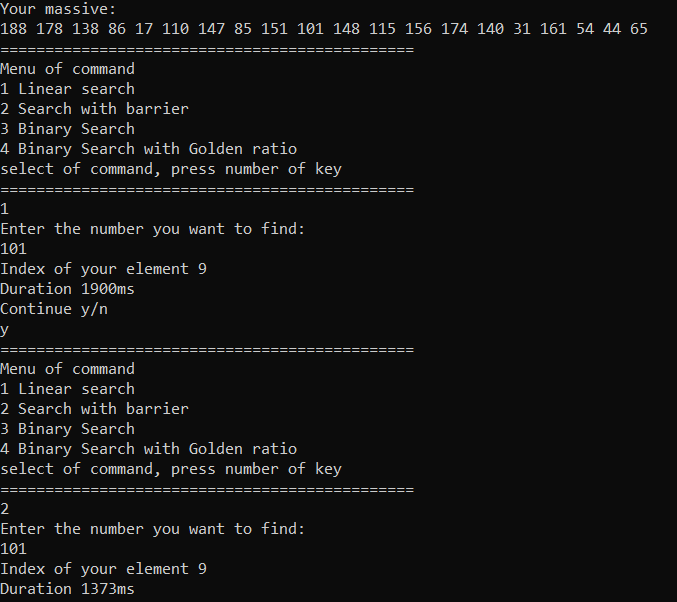
Масив на 10000 елементів: 81821 21495 24869 23047 77700 3174 64530 98654 45685 92501 61055 1261 38642 79540 73607 77476 25879 88832 12514 4912 7231 99130 74707 89844 56908 35796 32964 73012 29718 99186 4693 56872 179 45167 30447 4268 34594 12124 91334 4007 76873 57563 52737 65012 97535 99956 80796 29256 46058 9909 58829 37105 6038 33016 80225 82690 46788 35025 66139 41905 70209 91517 68937 46498 87809 4183 1082 93475 49093 69340 42882 13756 97497 8507 28039 49668 46394 9696 92308 56952 54999 3709 26955 19809 40358 67587 50834 67163 64941 47757 57480 99569 45985 49430 33798 85356 88625 6239 77202 93353 98874 87320 80162 76717 67027 45074 39608 97531 4977 30558 36297 37092 42716 9187 86905 66501 64562 49129 6141 20222 53349 33919 28535 1337 11860 43312 67771 48068 74709 42683 61042 40155 32493 98955 58077 67679 23546 50237 52781 95284 27980 53303 62186 34383 11460 20388 56853 36798 62524 67297 20794 39497 99 56981 40004 64956 58785 78826 64857 23716 77303 91540 22822 62294 69515 96141 4600 43761 51110 19227 42955 14325 96347 10857 92241 47 71732 94151 89876 91472 86458 30973 85543 7413 21889 21545 40057 75512 58951 93121 8894 44720 85380 29065 17981 36440 30481 99891 64867 15319 15789 13093 85688 43298 24342 6470 43149 89242 64739 39958 93223 64633 88950 73385 37257 46330 5997 15409 40406 47971 56084 29087 84810 57989 10333 98235 28945 67283 92875 55801 69565 71841 29285 74557 75683 73366 45285 42245 83070 15419 78395 50814 10773 18994 99897 44260 55770 11995 91808 71651 30443 15073 59485 16897 59234 86701 28282 27700 32965 26107 77525 75866 96367 8938 70392 21383 35349 14393 97701 63891 1045 63752 32339 24987 69575 5270 18315 65816 58091 66072 42465 16950 75474 21224 25359 54493 12356 12388 15324 88981 45084 17586 9280 56963 37894 2530 54447 1835 48601 78515 8912 41378 14293 27917 70606 66691 82734 34499 47322 53964 68387 62465 69608 66893 83642 60575 392 75143 83579 15900 9027 22961 99069 8721 18807 83459 54472 75375 87093 67046 50823 16480 17216 79576 57161 1088 2657 47558 50618 58667 71759 77857 65066 61359 20594 82618 47978 42387 73321 77318 2138 79304 18256 32701 69488 85950 42308 5573 9111 87147 12220 7359 47405 80276 3142 51908 45250 90032 9732 82554 57918 52834 24526 24912 44141 31170 57895 37711 88126 36188 35481 12085 2054 9775 64908 71518 98224 84515 10088 71210 58779 97084 38447 63470 46448 64400 30452 84790 58217 68686 37369 57945 32655 90767 19400 49303 54779 93343 88559 38318 28056 4598 17447 72922 51666 135 5306 37630 15368 31624 53684 60735 79944 38521 23775 60834 14388 61055 78464 79741 718 73105 79493 19440 98818 77536 79430 11286 90375 15606 16740 4881 95955 71769 43155 54181 79638 65524 11804 46314 59094 92822 69423 26594 15003 76634 78000 94133 66992 58566 34916 93723 14880 8819 37601 54951 25105 98006 74104 74130 20704 94020 47255 99761 14885 48803 64779 8176 95366 69593 81195 48863 49251 32150 20648 7896 3683 52900 4438 22184 99535 85296 44294 31809 80689 23016 90001 87137 34169 88204 29077 81631 91420 37674 25610 65074 45567 69661 11709 66199 11856 69825 98767 97399 77370 86053 44472 82729 88170 29705 51269 72204 93569 51920 74595 35408 49727 15446 19073 91868 58802 77642 74006 7162 40516 99564 57442 11578 6541 55573 62328 34021 41191 38238 10329 29710 86529 46894 54438 94636 78543 7273 98966 4464 36936 56873 88062 39501 63079 90202 15348 30121 50589 19840 7398 34213 95818 91605 56883 77806 28611 32894 76189 32597 67376 17248 31014 55331 41591 44885 48878 2833 61008 24436 13326 51529 78676 69542 70225 83643 11502 17942 48499 16338 40226 32208 83432 21351 30841 76116 33892 94923 90011 76632 66026 45649 31083 66369 37088 19689 25860 32731 97909 18195 41324 12516 69113 26153 6515 55868 55537 26276 21381 73239 66305 10669 84395 59851 92371 35169 33815 9663 24491 65475 10993 53955 12821 84375 58409 67677 20447 12307 32455 50537 57784 78770 20033 30305 75013 27711 63094 57280 14837 74972 20579 7616 73543 16773 10328 55358 61254 71231 73998 1919 85874 16197 8241 32435 7202 87368 99694 84738 67744 38838 35422 43230 93821 70843 15456 87534 90636 54090 84837 64660 72075 76535 94826 49519 44896 3378 80412 96048 74048 3052 6423 49215 38309 30536 64663 41587 70530 87598 23104 87645 28016 78356 89535 75999 39550 77348 13794 2700 1318 22793 64722 7164 86394 9338 29844 9339 21371 58722 87538 3824 6956 3647 10690 64692 32415 86207 4921**…..**59191 26497 96397 9641 19284 11982 48266 77128 22079 74271 29452 91409 50243 53655 77308 74933 13454 90181 54781 67724 50469 99318 22783 69880 54337 81666 34132 4179 55857 22442 49320 58860 25511 36830 24148 20152 83590 61682 86200 70161 73215 62713 68136 55356 56108 98195 60385 35071 63546 74881 44432 66895 14562 30797 78091 59873 3714 26517 55304 37618 77849 44087 21271 99636 24951 70591 65899 13413 29508 57156 91342 51772 3982 84619 94509 87755 31183 67427 13772 56141 21281 99060 40633 80975 78010 82425 44299 94713 58738 64275 16439 57074 6379 48583 92311 29128 41443 68499 30405 85516 32297 46973 5564 6390 83539 92660 62913 29942 36288 90336 28690 36286 12745 99162 16337 78788 3455 40295 21627 42526 26293 71185 54675 65233 40718 34269 45394 55399 92307 65382 46312 62684 37022 28256 23845 74308 93497 34243 97436 85350 19512 14357 58426 68402 35585 80152 44329 7216 32246 13524 25673 1148 46873 87879 59224 61111 92298 40330 65672 49270 47267 7515 37634 70905 44024 66006 99264 84489 75318 42029 85291 69212 42859 36276 23015 11937 12760 86272 29648 60565 89939 38178 23153 78784 39709 45165 15811 49143 69031 62733 82010 53167 68571 48167 38084 11997 76722 87520 24378 69579 80888 59840 31898 90216 28233 56461 58289 10597 64865 47287 48352 6026 10683 5108 59330 9089 84362 58850 31197 98859 50195 50121 15458 12585 60019 32124 642 94691 98192 11018 35215 25239 61894 60069 79677 30100 6281 32936 50552 83248 86875 5214 84278 21745 14381 75984 42485 63464 88836 28995 27635 17873 56324 25519 19384 30767 44382 19313 30198 96108 55820 63597 12405 53847 12073 23114 44131 50614 25430 29231 2809 51426 8298 47948 11984 44982 65843 93713 98203 84034 89638 37178 5856 69197 83383 17343 68914 36376 40685 48910 85717 86967 2739 54444 27428 23279 92809 30432 9673 91267 31854 91871 38034 59606 26187 76209 9575 8376 13536 74402 64924 50669 21110 46164 6471 18643 57742 43207 70463 94730 54200 7756 9929 76520 43513 15842 48688 53854 20521 88705 52172 35412 84041 6937 52181 97767 99143 46249 43011 7175 7768 60538 22841 66282 83986 35049 65858 56575 47973 8785 65537 49602 59968 14942 37068 24098 81942 61513 16094 10346 27522 55721 87855 24831 22231 29512 66628 14173 93982 8704 19499 11493 196 63289 86962 93661 84914 43647 92534 8464 49983 92112 5881 11736 23168 2570 75444 69099 69868 28083 15825 37631 30156 32666 79653 52046 72682 98010 44052 54474 68420 51685 62592 39270 89286 54044 49407 96678 27980 43406 39184 38296 68979 7630 60362 35409 49841 76120 1663 19661 54068 97529 44305 65900 92879 81868 81336 67530 83636 34917 20979 86904 99594

# **Результати тестування:**

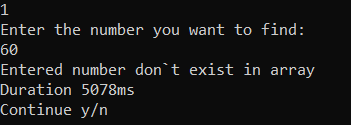
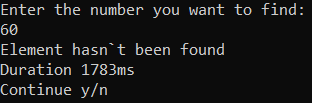
Тест №1:

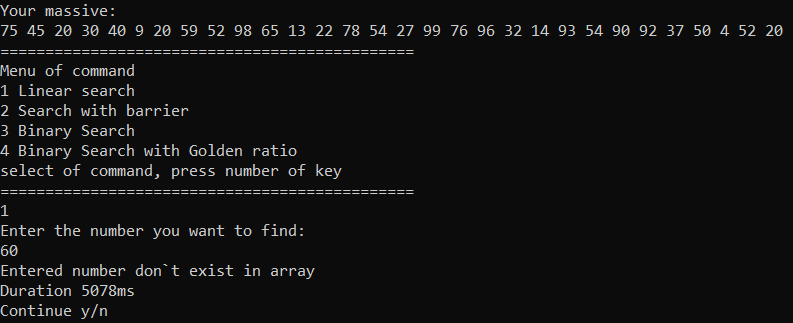
Linear Search: Barrier Search:

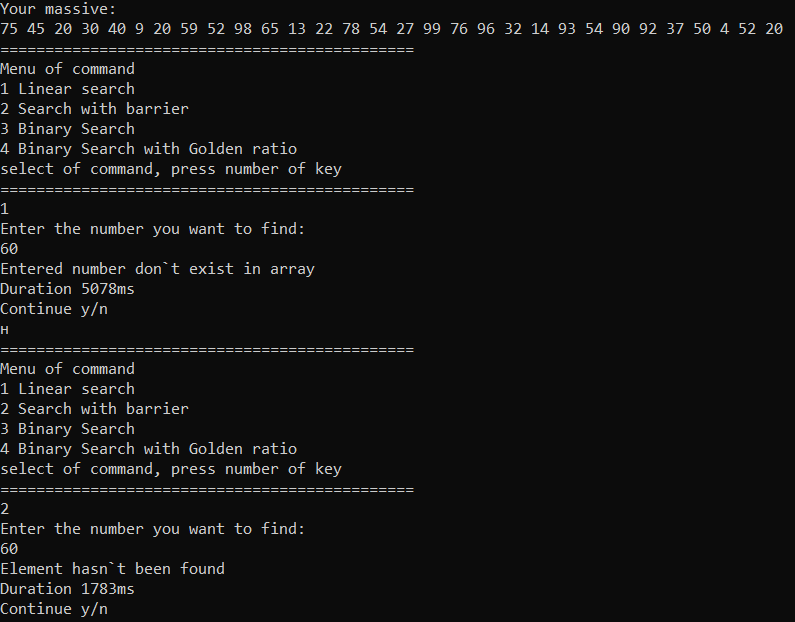
Linear Search:

Barrier Search:

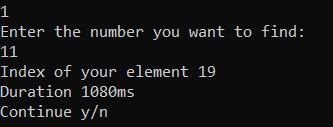
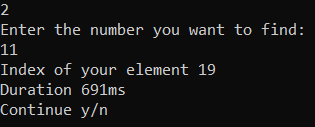
Тест №2:

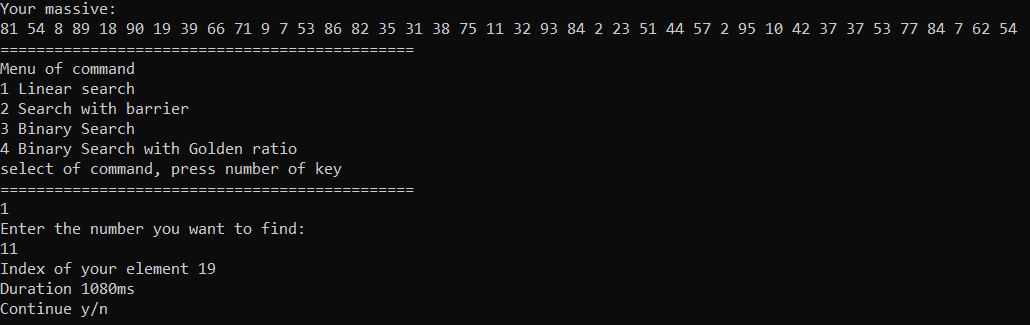
Linear Search: Barrier Search:

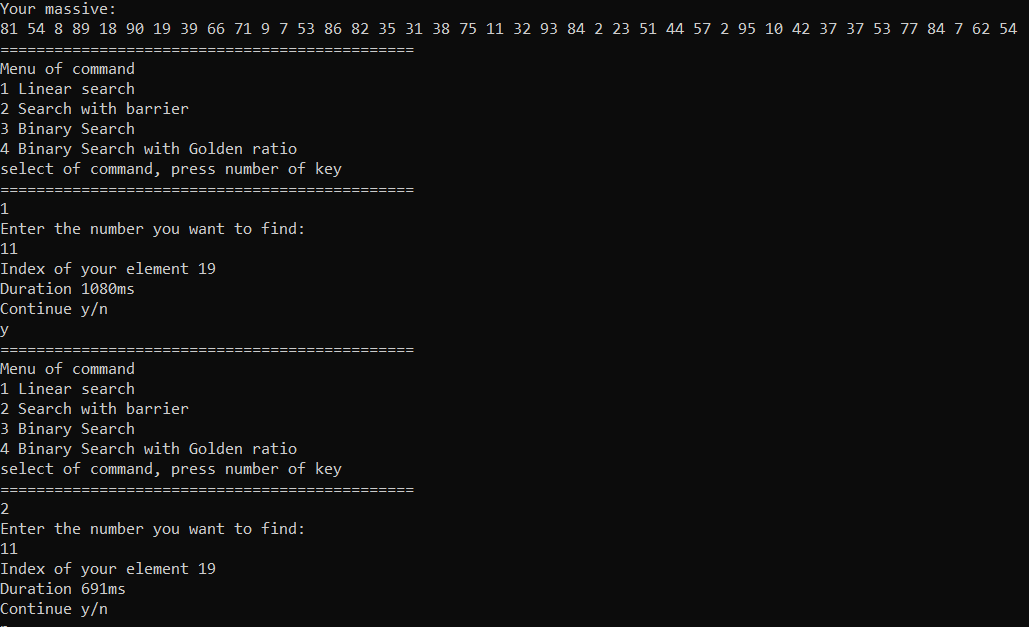
Linear Search:

Barrier Search:

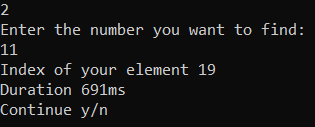
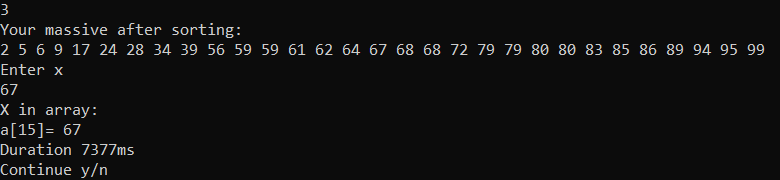
Тест №3:

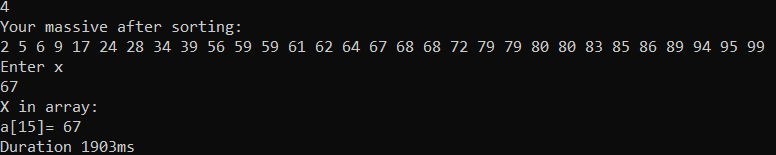
Linear Search: Barrier Search:

Linear Search:

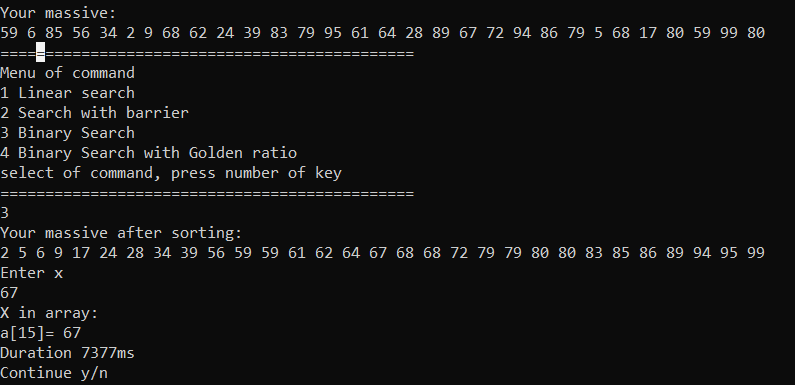
Barrier Search:

Тест №4:

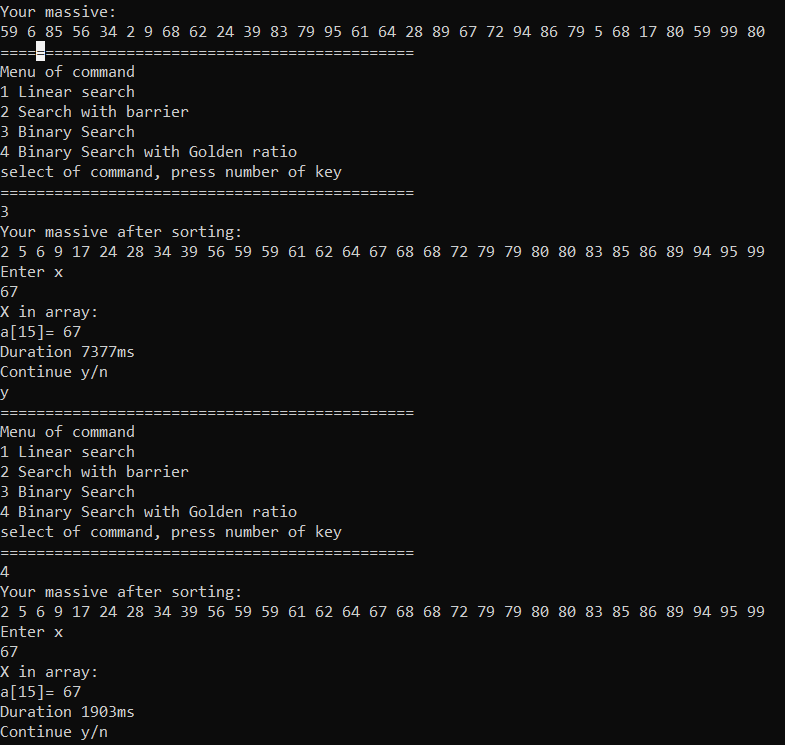
Binary Search: Binary Search with Golden ratio rule:



Binary Search:



Binary Search with Golden ratio rule:



У фінальному варіанті програми не виявлено жодних помилок під час тестувань, отже усі алгоритму працюють правильно, видають очікувані результати з приводу часу та ефективності.

В процесі написання коду були численні помилки з бінарним пошуком на зв’язному списку(з золотим перерізом та без нього), усе це через неправильно задану праву границю пошуку, адже останній елемент списку має адресу наступного, неіснуючого, тобто нульову адресу, але має. І через це я не міг розібратися, допоки не збільшив на 1 праву границю.