



# DATA STRUCTURES & ALGORITHMS

**CS106.3**

**Coursework 2023/2024**

<b>Name</b>	<b>DCK Gunawardhane</b>
<b>Index</b>	<b>26944</b>
<b>Degree</b>	<b>BSc (Hons)Technology Management</b>

## Coursework

Use recursive approach to solve the Binary Search:

1. Write a proper pseudocode
2. Convert in to a program
3. Test the program with [2, 8, 9, 11, 15, 45, 58, 78, 99] array locating 2.

----- [Answer Below] -----

### Pseudo code

```
Function recursiveBinarySearch (ar, low, max, target)
  If low > max
    Return -1

  mid = (low + max) / 2
  midValue = ar[mid]

  If midValue == target
    Return mid // Base case: target found at index mid

  If target < midValue
    Return recursiveBinarySearch (ar, low, mid - 1, target)
  Else
    Return recursiveBinarySearch (ar, mid + 1, max, target)
  End If
End Function
```

## C code

```
if (low > max)
    return -1;

int mid = (low + max) / 2;
int midValue = ar[mid];

if (midValue == target)
    return mid;

if (target < midValue)
    return recursiveBinarySearch(ar, low, mid - 1, target);
else
    return recursiveBinarySearch(ar, mid + 1, max, target);
}

int main() {
    int ar[] = {2, 8, 9, 11, 15, 45, 58, 78, 99};
    int n = sizeof(ar) / sizeof(ar[0]);
    int target = 2;

    int result = recursiveBinarySearch(ar, 0, n - 1, target);

    if (result != -1)
        printf("Target %d found at index %d.\n", target, result);
    else
        printf("Target %d not found in the array.\n", target);

    return 0;
}
```

```
1  algo assignment 27660.c - Code::Blocks 20.03
2  File Edit View Search Project Build Debug Fortran wxSmith Tools Tools - Plugins DoxyBlocks Settings Help
3  <global>
4  Start here x algo assignment 27660.c x
5  4 if (low > max)
6     return -1;
7
8  5 int mid = (low + max) / 2;
9   int midValue = ar[mid];
10
11  6 if (midValue == target)
12     return mid;
13
14  7 if (target < midValue)
15     return recursiveBinarySearch(ar, low, mid - 1, target);
16  8 else
17     return recursiveBinarySearch(ar, mid + 1, max, target);
18
19  9
20  10
21  11
22  12
23  13
24  14
25  15
26  16
27  17
28  18
29  19
30  20 int main() {
31  21     int ar[] = {2, 0, 0, 11, 15, 45, 50, 70, 99};
32  22     int n = sizeof(ar) / sizeof(ar[0]);
33  23     int target = 2;
34
35  24     int result = recursiveBinarySearch(ar, 0, n - 1, target);
36
37  25     if (result != -1)
38  26         printf("Target %d found at index %d.\n", target, result);
39  27     else
40  28         printf("Target %d not found in the array.\n", target);
41
42  29     return 0;
43  30 }
44
45  31
46  32
47  33
48  34
49  35
50  36
51  37
52  38
53  39
54  40
55  41
56  42
57  43
58  44
59  45
60  46
61  47
62  48
63  49
64  50
65  51
66  52
67  53
68  54
69  55
70  56
71  57
72  58
73  59
74  60
75  61
76  62
77  63
78  64
79  65
80  66
81  67
82  68
83  69
84  70
85  71
86  72
87  73
88  74
89  75
90  76
91  77
92  78
93  79
94  80
95  81
96  82
97  83
98  84
99  85
100 86
101 87
102 88
103 89
104 90
105 91
106 92
107 93
108 94
109 95
110 96
111 97
112 98
113 99
114 100
115 101
116 102
117 103
118 104
119 105
120 106
121 107
122 108
123 109
124 110
125 111
126 112
127 113
128 114
129 115
130 116
131 117
132 118
133 119
134 120
135 121
136 122
137 123
138 124
139 125
140 126
141 127
142 128
143 129
144 130
145 131
146 132
147 133
148 134
149 135
150 136
151 137
152 138
153 139
154 140
155 141
156 142
157 143
158 144
159 145
160 146
161 147
162 148
163 149
164 150
165 151
166 152
167 153
168 154
169 155
170 156
171 157
172 158
173 159
174 160
175 161
176 162
177 163
178 164
179 165
180 166
181 167
182 168
183 169
184 170
185 171
186 172
187 173
188 174
189 175
190 176
191 177
192 178
193 179
194 180
195 181
196 182
197 183
198 184
199 185
200 186
201 187
202 188
203 189
204 190
205 191
206 192
207 193
208 194
209 195
210 196
211 197
212 198
213 199
214 200
215 201
216 202
217 203
218 204
219 205
220 206
221 207
222 208
223 209
224 210
225 211
226 212
227 213
228 214
229 215
230 216
231 217
232 218
233 219
234 220
235 221
236 222
237 223
238 224
239 225
240 226
241 227
242 228
243 229
244 230
245 231
246 232
247 233
248 234
249 235
250 236
251 237
252 238
253 239
254 240
255 241
256 242
257 243
258 244
259 245
260 246
261 247
262 248
263 249
264 250
265 251
266 252
267 253
268 254
269 255
270 256
271 257
272 258
273 259
274 260
275 261
276 262
277 263
278 264
279 265
280 266
281 267
282 268
283 269
284 270
285 271
286 272
287 273
288 274
289 275
290 276
291 277
292 278
293 279
294 280
295 281
296 282
297 283
298 284
299 285
300 286
301 287
302 288
303 289
304 290
305 291
306 292
307 293
308 294
309 295
310 296
311 297
312 298
313 299
314 300
315 301
316 302
317 303
318 304
319 305
320 306
321 307
322 308
323 309
324 310
325 311
326 312
327 313
328 314
329 315
330 316
331 317
332 318
333 319
334 320
335 321
336 322
337 323
338 324
339 325
340 326
341 327
342 328
343 329
344 330
345 331
346 332
347 333
348 334
349 335
350 336
351 337
352 338
353 339
354 340
355 341
356 342
357 343
358 344
359 345
360 346
361 347
362 348
363 349
364 350
365 351
366 352
367 353
368 354
369 355
370 356
371 357
372 358
373 359
374 360
375 361
376 362
377 363
378 364
379 365
380 366
381 367
382 368
383 369
384 370
385 371
386 372
387 373
388 374
389 375
390 376
391 377
392 378
393 379
394 380
395 381
396 382
397 383
398 384
399 385
400 386
401 387
402 388
403 389
404 390
405 391
406 392
407 393
408 394
409 395
410 396
411 397
412 398
413 399
414 400
415 401
416 402
417 403
418 404
419 405
420 406
421 407
422 408
423 409
424 410
425 411
426 412
427 413
428 414
429 415
430 416
431 417
432 418
433 419
434 420
435 421
436 422
437 423
438 424
439 425
440 426
441 427
442 428
443 429
444 430
445 431
446 432
447 433
448 434
449 435
450 436
451 437
452 438
453 439
454 440
455 441
456 442
457 443
458 444
459 445
460 446
461 447
462 448
463 449
464 450
465 451
466 452
467 453
468 454
469 455
470 456
471 457
472 458
473 459
474 460
475 461
476 462
477 463
478 464
479 465
480 466
481 467
482 468
483 469
484 470
485 471
486 472
487 473
488 474
489 475
490 476
491 477
492 478
493 479
494 480
495 481
496 482
497 483
498 484
499 485
500 486
501 487
502 488
503 489
504 490
505 491
506 492
507 493
508 494
509 495
510 496
511 497
512 498
513 499
514 500
515 501
516 502
517 503
518 504
519 505
520 506
521 507
522 508
523 509
524 510
525 511
526 512
527 513
528 514
529 515
530 516
531 517
532 518
533 519
534 520
535 521
536 522
537 523
538 524
539 525
540 526
541 527
542 528
543 529
544 530
545 531
546 532
547 533
548 534
549 535
550 536
551 537
552 538
553 539
554 540
555 541
556 542
557 543
558 544
559 545
560 546
561 547
562 548
563 549
564 550
565 551
566 552
567 553
568 554
569 555
570 556
571 557
572 558
573 559
574 560
575 561
576 562
577 563
578 564
579 565
580 566
581 567
582 568
583 569
584 570
585 571
586 572
587 573
588 574
589 575
590 576
591 577
592 578
593 579
594 580
595 581
596 582
597 583
598 584
599 585
600 586
601 587
602 588
603 589
604 590
605 591
606 592
607 593
608 594
609 595
610 596
611 597
612 598
613 599
614 600
615 601
616 602
617 603
618 604
619 605
620 606
621 607
622 608
623 609
624 610
625 611
626 612
627 613
628 614
629 615
630 616
631 617
632 618
633 619
634 620
635 621
636 622
637 623
638 624
639 625
640 626
641 627
642 628
643 629
644 630
645 631
646 632
647 633
648 634
649 635
650 636
651 637
652 638
653 639
654 640
655 641
656 642
657 643
658 644
659 645
660 646
661 647
662 648
663 649
664 650
665 651
666 652
667 653
668 654
669 655
670 656
671 657
672 658
673 659
674 660
675 661
676 662
677 663
678 664
679 665
680 666
681 667
682 668
683 669
684 670
685 671
686 672
687 673
688 674
689 675
690 676
691 677
692 678
693 679
694 680
695 681
696 682
697 683
698 684
699 685
700 686
701 687
702 688
703 689
704 690
705 691
706 692
707 693
708 694
709 695
710 696
711 697
712 698
713 699
714 700
715 701
716 702
717 703
718 704
719 705
720 706
721 707
722 708
723 709
724 710
725 711
726 712
727 713
728 714
729 715
730 716
731 717
732 718
733 719
734 720
735 721
736 722
737 723
738 724
739 725
740 726
741 727
742 728
743 729
744 730
745 731
746 732
747 733
748 734
749 735
750 736
751 737
752 738
753 739
754 740
755 741
756 742
757 743
758 744
759 745
760 746
761 747
762 748
763 749
764 750
765 751
766 752
767 753
768 754
769 755
770 756
771 757
772 758
773 759
774 760
775 761
776 762
777 763
778 764
779 765
780 766
781 767
782 768
783 769
784 770
785 771
786 772
787 773
788 774
789 775
790 776
791 777
792 778
793 779
794 780
795 781
796 782
797 783
798 784
799 785
800 786
801 787
802 788
803 789
804 790
805 791
806 792
807 793
808 794
809 795
810 796
811 797
812 798
813 799
814 800
815 801
816 802
817 803
818 804
819 805
820 806
821 807
822 808
823 809
824 810
825 811
826 812
827 813
828 814
829 815
830 816
831 817
832 818
833 819
834 820
835 821
836 822
837 823
838 824
839 825
840 826
841 827
842 828
843 829
844 830
845 831
846 832
847 833
848 834
849 835
850 836
851 837
852 838
853 839
854 840
855 841
856 842
857 843
858 844
859 845
860 846
861 847
862 848
863 849
864 850
865 851
866 852
867 853
868 854
869 855
870 856
871 857
872 858
873 859
874 860
875 861
876 862
877 863
878 864
879 865
880 866
881 867
882 868
883 869
884 870
885 871
886 872
887 873
888 874
889 875
890 876
891 877
892 878
893 879
894 880
895 881
896 882
897 883
898 884
899 885
900 886
901 887
902 888
903 889
904 890
905 891
906 892
907 893
908 894
909 895
910 896
911 897
912 898
913 899
914 900
915 901
916 902
917 903
918 904
919 905
920 906
921 907
922 908
923 909
924 910
925 911
926 912
927 913
928 914
929 915
930 916
931 917
932 918
933 919
934 920
935 921
936 922
937 923
938 924
939 925
940 926
941 927
942 928
943 929
944 930
945 931
946 932
947 933
948 934
949 935
950 936
951 937
952 938
953 939
954 940
955 941
956 942
957 943
958 944
959 945
960 946
961 947
962 948
963 949
964 950
965 951
966 952
967 953
968 954
969 955
970 956
971 957
972 958
973 959
974 960
975 961
976 962
977 963
978 964
979 965
980 966
981 967
982 968
983 969
984 970
985 971
986 972
987 973
988 974
989 975
990 976
991 977
992 978
993 979
994 980
995 981
996 982
997 983
998 984
999 985
1000 986
1001 987
1002 988
1003 989
1004 990
1005 991
1006 992
1007 993
1008 994
1009 995
1010 996
1011 997
1012 998
1013 999
1014 1000
1015 1001
1016 1002
1017 1003
1018 1004
1019 1005
1020 1006
1021 1007
1022 1008
1023 1009
1024 1010
1025 1011
1026 1012
1027 1013
1028 1014
1029 1015
1030 1016
1031 1017
1032 1018
1033 1019
1034 1020
1035 1021
1036 1022
1037 1023
1038 1024
1039 1025
1040 1026
1041 1027
1042 1028
1043 1029
1044 1030
1045 1031
1046 1032
1047 1033
1048 1034
1049 1035
1050 1036
1051 1037
1052 1038
1053 1039
1054 1040
1055 1041
1056 1042
1057 1043
1058 1044
1059 1045
1060 1046
1061 1047
1062 1048
1063 1049
1064 1050
1065 1051
1066 1052
1067 1053
1068 1054
1069 1055
1070 1056
1071 1057
1072 1058
1073 1059
1074 1060
1075 1061
1076 1062
1077 1063
1078 1064
1079 1065
1080 1066
1081 1067
1082 1068
1083 1069
1084 1070
1085 1071
1086 1072
1087 1073
1088 1074
1089 1075
1090 1076
1091 1077
1092 1078
1093 1079
1094 1080
1095 1081
1096 1082
1097 1083
1098 1084
1099 1085
1100 1086
1101 1087
1102 1088
1103 1089
1104 1090
1105 1091
1106 1092
1107 1093
1108 1094
1109 1095
1110 1096
1111 1097
1112 1098
1113 1099
1114 1100
1115 1101
1116 1102
1117 1103
1118 1104
1119 1105
1120 1106
1121 1107
1122 1108
1123 1109
1124 1110
1125 1111
1126 1112
1127 1113
1128 1114
1129 1115
1130 1116
1131 1117
1132 1118
1133 1119
1134 1120
1135 1121
1136 1122
1137 1123
1138 1124
1139 1125
1140 1126
1141 1127
1142 1128
1143 1129
1144 1130
1145 1131
1146 1132
1147 1133
1148 1134
1149 1135
1150 1136
1151 1137
1152 1138
1153 1139
1154 1140
1155 1141
1156 1142
1157 1143
1158 1144
1159 1145
1160 1146
1161 1147
1162 1148
1163 1149
1164 1150
1165 1151
1166 1152
1167 1153
1168 1154
1169 1155
1170 1156
1171 1157
1172 1158
1173 1159
1174 1160
1175 1161
1176 1162
1177 1163
1178 1164
1179 1165
1180 1166
1181 1167
1182 1168
1183 1169
1184 1170
1185 1171
1186 1172
1187 1173
1188 1174
1189 1175
1190 1176
1191 1177
1192 1178
1193 1179
1194 1180
1195 1181
1196 1182
1197 1183
1198 1184
1199 1185
1200 1186
1201 1187
1202 1188
1203 1189
1204 1190
1205 1191
1206 1192
1207 1193
1208 1194
1209 1195
1210 1196
1211 1197
1212 1198
1213 1199
1214 1200
1215 1201
1216 1202
1217 1203
1218 1204
1219 1205
1220 1206
1221 1207
1222 1208
1223 1209
1224 1210
1225 1211
1226 1212
1227 1213
1228 1214
1229 1215
1230 1216
1231 1217
1232 1218
1233 1219
1234 1220
1235 1221
1236 1222
1237 1223
1238 1224
1239 1225
1240 1226
1241 1227
1242 1228
1243 1229
1244 1230
1245 1231
1246 1232
1247 1233
1248 1234
1249 1235
1250 1236
1251 1237
1252 1238
1253 1239
1254 1240
1255 1241
1256 1242
1257 1243
1258 1244
1259 1245
1260 1246
1261 1247
1262 1248
1263 1249
1264 1250
1265 1251
1266 1252
1267 1253
1268 1254
1269 1255
1270 1256
1271 1257
1272 1258
1273 1259
1274 1260
1275 1261
1276 1262
1277 1263
1278 1264
1279 1265
1280 1266
1281 1267
1282 1268
1283 1269
1284 1270
1285 1271
1286 1272
1287 1273
1288 1274
1289 1275
1290 1276
1291 1277
1292 1278
1293 1279
1294 1280
1295 1281
1296 1282
1297 1283
1298 1284
1299 1285
1300 1286
1301 1287
1302 1288
1303 1289
1304 1290
1305 1291
1306 1292
1307 1293
1308 1294
1309 1295
1310 1296
1311 1297
1312 1298
1313 1299
1314 1300
1315 1301
1316 1302
1317 1303
1318 1304
1319 1305
1320 1306
1321 1307
1322 1308
1323 1309
1324 1310
1325 1311
1326 1312
1327 1313
1328 1314
1329 1315
1330 1316
1331 1317
1332 1318
1333 1319
1334 1320
1335 1321
1336 1322
1337 1323
1338 1324
1339 1325
1340 1326
1341 1327
1342 1328
1343 1329
1344 1330
1345 1331
1346 1332
1347 1333
1348 1334
1349 1335
1350 1336
1351 1337
1352 1338
1353 1339
1354 1340
1355 1341
1356 1342
1357 1343
1358 1344
1359 1345
1360 1346
1361 1347
1362 1348
1363 1349
1364 1350
1365 1351
1366 1352
1367 1353
1368 1354
1369 1355
1370 1356
1371 1357
1372 1358
1373 1359
1374 1360
1375 1361
1376 1362
1377 1363
1378 1364
1379 1365
1380 1366
1381 1367
1382 1368
1383 1369
1384 1370
1385 1371
1386 1372
1387 1373
1388 1374
1389 1375
1390 1376
1391 1377
1392 1378
1393 1379
1394 1380
1395 1381
1396 1382
1397 1383
1398 1384
1399 1385
1400 1386
1401 1387
1402 1388
1403 1389
1404 1390
1405 1391
1406 1392
1407 1393
1408 1394
1409 1395
1410 1396
1411 1397
1412 1398
1413 1399
1414 1400
1415 1401
1416 1402
1417 1403
1418 1404
1419 1405

```