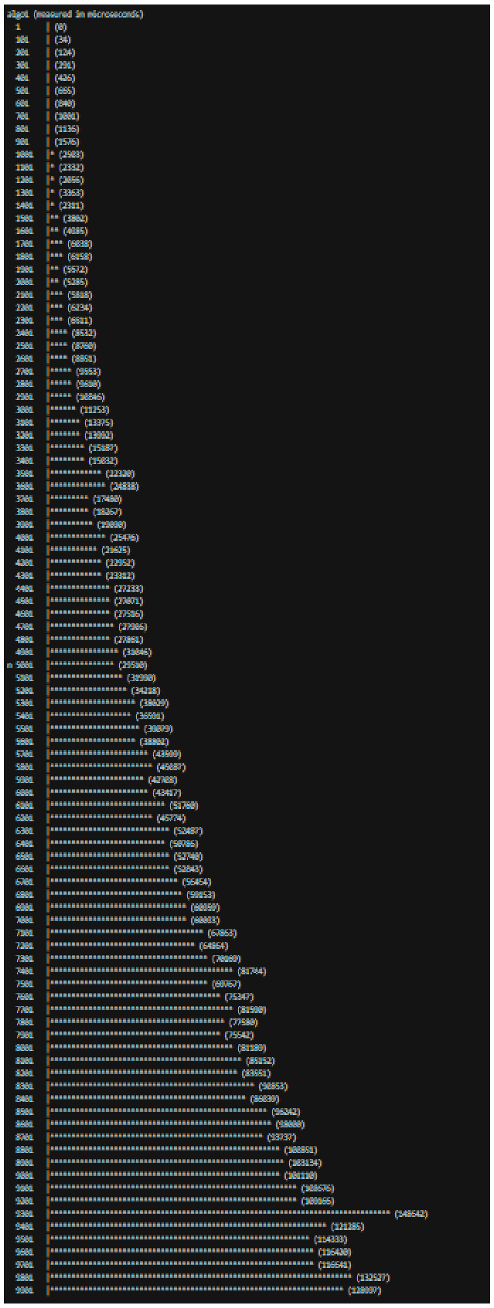
LE 2

Algorithm 1:

Algorithm one has a time complexity of O(n^2). This is because algorithm one has 2 nested for loops which can be generalized into a polynomial runtime of n^2. Another way it can be found is based on the behavior of the graph below which looks similar to the graph of n^2. The reason behind this is that the nested for loop goes through a vector of size n to find the count of sets of 2 numbers that have a sum of “target” which would result in the size being crossed by itself in the process of searching for these sets.

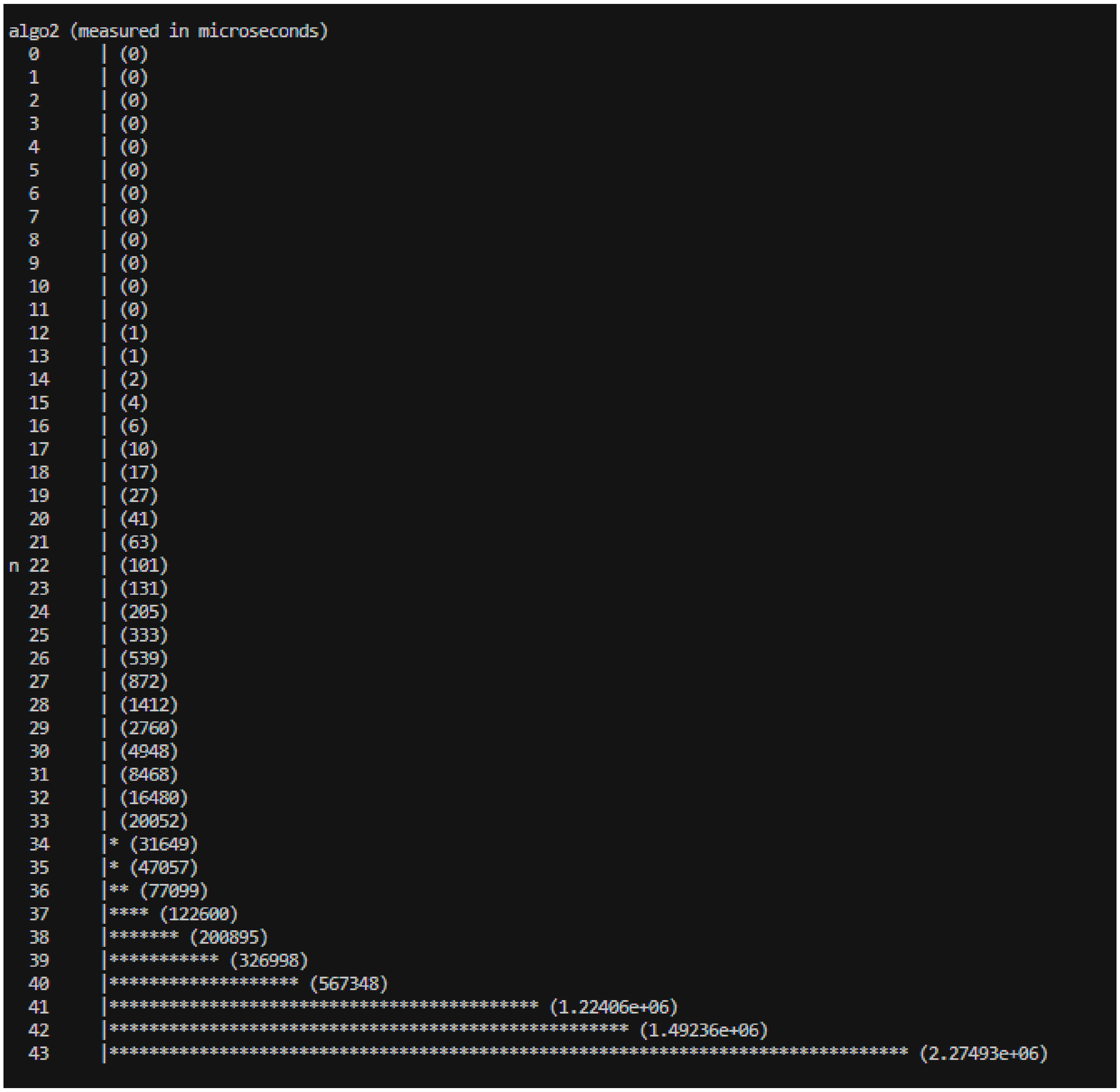
Algorithm 1 Graph -



```
algo1 (measured in microseconds)
  1     | (0)
  101   | (34)
  201   | (124)
  301   | (291)
  401   | (426)
  501   | (665)
  601   | (840)
  701   | (1081)
  801   | (1136)
  901   | (1576)
  1001  |* (2003)
  1101  |* (2332)
  1201  |* (2856)
  1301  |* (3363)
  1401  |* (2311)
  1501  |** (3802)
  1601  |** (4085)
  1701  |*** (6038)
  1801  |*** (6158)
  1901  |** (5572)
  2001  |** (5285)
  2101  |*** (5818)
  2201  |*** (6234)
  2301  |*** (6511)
  2401  |**** (8532)
  2501  |**** (8760)
  2601  |**** (8851)
  2701  |***** (9553)
  2801  |***** (9610)
  2901  |***** (10846)
  3001  |****** (11253)
  3101  |******* (13375)
  3201  |******* (13092)
  3301  |******** (15187)
  3401  |******** (15832)
  3501  |************ (22320)
  3601  |************* (24838)
  3701  |********* (17480)
  3801  |********* (18267)
  3901  |********** (19800)
  4001  |************* (25476)
  4101  |*********** (21625)
  4201  |************ (22952)
  4301  |************ (23312)
  4401  |************** (27233)
  4501  |************** (27871)
  4601  |************** (27516)
  4701  |*************** (27986)
  4801  |************** (27861)
  4901  |**************** (30846)
n 5001  |*************** (29518)
  5101  |***************** (31908)
  5201  |****************** (34218)
  5301  |******************** (38629)
  5401  |******************* (36591)
  5501  |********************* (39079)
  5601  |******************** (38802)
  5701  |*********************** (43909)
  5801  |************************ (45887)
  5901  |********************** (42708)
  6001  |*********************** (43417)
  6101  |*************************** (51760)
  6201  |************************ (45774)
  6301  |**************************** (52487)
  6401  |*************************** (50786)
  6501  |**************************** (52740)
  6601  |**************************** (52843)
  6701  |****************************** (56454)
  6801  |******************************* (58153)
  6901  |******************************** (60059)
  7001  |******************************** (60803)
  7101  |************************************ (67863)
  7201  |********************************** (64864)
  7301  |************************************* (70160)
  7401  |******************************************* (81744)
  7501  |************************************* (69767)
  7601  |**************************************** (75347)
  7701  |******************************************* (81508)
  7801  |***************************************** (77580)
  7901  |**************************************** (75542)
  8001  |******************************************* (81180)
  8101  |********************************************* (85152)
  8201  |******************************************** (83551)
  8301  |************************************************ (90853)
  8401  |********************************************** (86830)
  8501  |*************************************************** (96242)
  8601  |**************************************************** (98080)
  8701  |************************************************** (93737)
  8801  |****************************************************** (100851)
  8901  |******************************************************* (103134)
  9001  |****************************************************** (101118)
  9101  |********************************************************** (108576)
  9201  |********************************************************** (108165)
  9301  |******************************************************************************** (148542)
  9401  |***************************************************************** (121285)
  9501  |************************************************************* (114333)
  9601  |************************************************************** (116420)
  9701  |************************************************************** (116541)
  9801  |*********************************************************************** (132527)
  9901  |********************************************************************* (128097)
```

__**Algorithm 2:**__
Algorithm two has a time complexity of O(2^n). This is because algorithm two is implemented using recursion where algorithm 2 is being implemented on all numbers between n and 2 when n is greater than 2 and since a number can either be in or not in the subset of numbers greater than 2, the total subsets would be 2^n and would result in a time complexity of O(2^n). Furthermore, the graph of algorithm 2 is a fast growing graph and also behaves like the function 2^n.

Algorithm 2 Graph -

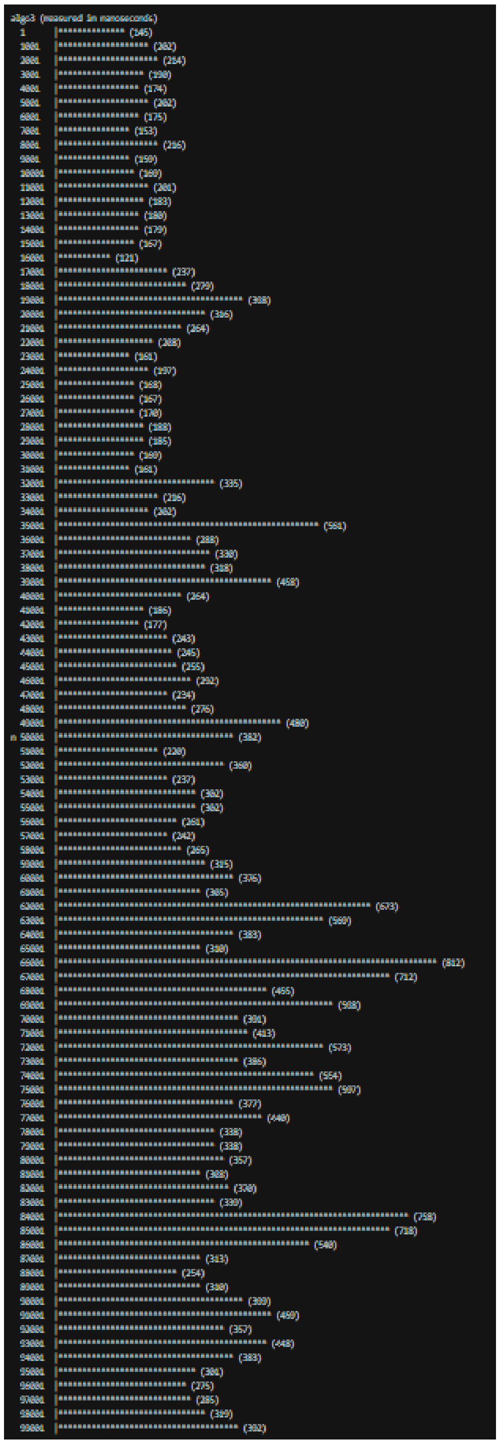


```
algo2 (measured in microseconds)
  0      | (0)
  1      | (0)
  2      | (0)
  3      | (0)
  4      | (0)
  5      | (0)
  6      | (0)
  7      | (0)
  8      | (0)
  9      | (0)
  10     | (0)
  11     | (0)
  12     | (1)
  13     | (1)
  14     | (2)
  15     | (4)
  16     | (6)
  17     | (10)
  18     | (17)
  19     | (27)
  20     | (41)
  21     | (63)
n 22     | (101)
  23     | (131)
  24     | (205)
  25     | (333)
  26     | (539)
  27     | (872)
  28     | (1412)
  29     | (2760)
  30     | (4948)
  31     | (8468)
  32     | (16480)
  33     | (20052)
  34     |* (31649)
  35     |* (47057)
  36     |** (77099)
  37     |**** (122600)
  38     |******* (200895)
  39     |*********** (326998)
  40     |******************* (567348)
  41     |******************************************* (1.22406e+06)
  42     |**************************************************** (1.49236e+06)
  43     |******************************************************************************** (2.27493e+06)
```

Algorithm 3:

Algorithm three has a time complexity of O(logn). This is because algorithm 3 uses recursion in order to find a “target” number in the vector “lis_arr” (searching algorithm) and the algorithm does this by dividing the sum of “high” and “low” by 2 in order to find the midpoint of the 2 indices and this is a good indication for the run time complexity to be O(logn). The graph of algorithm 3 also grows at a slow rate which is identical to the function logn.

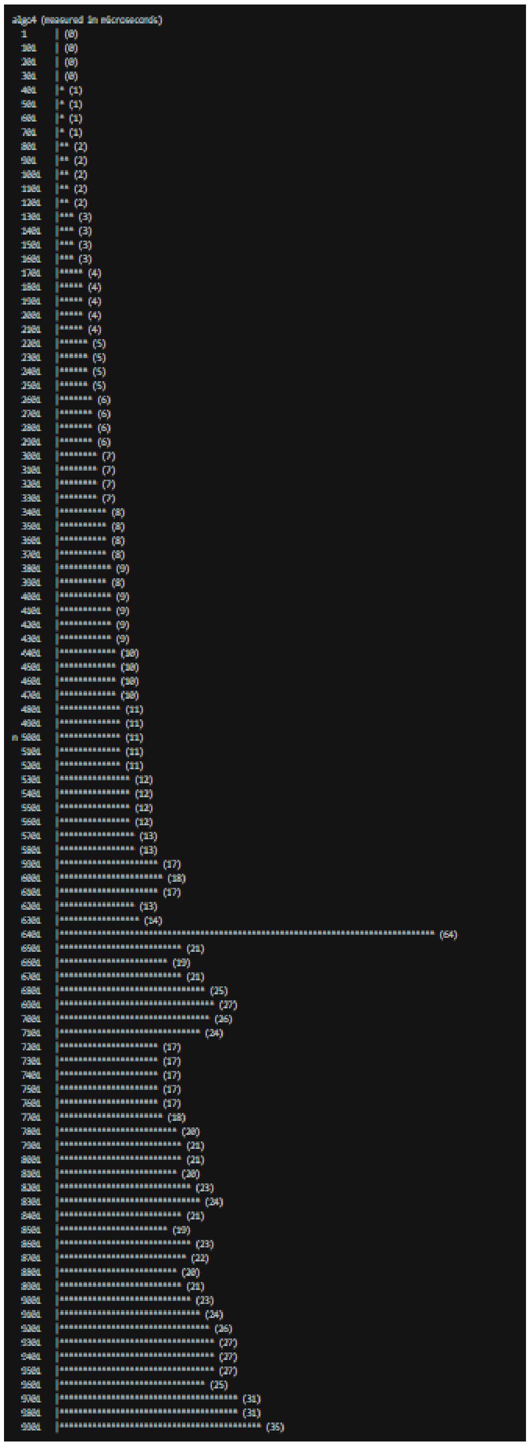
Algorithm 3 Graph -



```
algo3 (measured in nanoseconds)
  1      |************** (145)
  1001   |******************* (202)
  2001   |********************* (214)
  3001   |****************** (190)
  4001   |***************** (174)
  5001   |******************* (202)
  6001   |***************** (175)
  7001   |*************** (153)
  8001   |********************* (216)
  9001   |*************** (159)
  10001  |**************** (169)
  11001  |******************* (201)
  12001  |****************** (183)
  13001  |***************** (180)
  14001  |***************** (179)
  15001  |**************** (167)
  16001  |*********** (121)
  17001  |*********************** (237)
  18001  |*************************** (279)
  19001  |*************************************** (398)
  20001  |******************************* (316)
  21001  |************************** (264)
  22001  |******************** (208)
  23001  |*************** (161)
  24001  |******************* (197)
  25001  |**************** (168)
  26001  |**************** (167)
  27001  |**************** (170)
  28001  |****************** (188)
  29001  |****************** (185)
  30001  |**************** (169)
  31001  |*************** (161)
  32001  |********************************* (335)
  33001  |********************* (216)
  34001  |******************* (202)
  35001  |******************************************************* (561)
  36001  |**************************** (288)
  37001  |******************************** (330)
  38001  |******************************* (318)
  39001  |********************************************* (458)
  40001  |************************** (264)
  41001  |****************** (186)
  42001  |***************** (177)
  43001  |*********************** (243)
  44001  |************************ (245)
  45001  |************************* (255)
  46001  |**************************** (292)
  47001  |*********************** (234)
  48001  |*************************** (276)
  49001  |*********************************************** (488)
n 50001  |************************************* (382)
  51001  |********************* (220)
  52001  |*********************************** (360)
  53001  |*********************** (237)
  54001  |***************************** (302)
  55001  |***************************** (302)
  56001  |************************* (261)
  57001  |*********************** (242)
  58001  |************************** (265)
  59001  |******************************* (315)
  60001  |************************************* (376)
  61001  |****************************** (305)
  62001  |****************************************************************** (673)
  63001  |******************************************************** (569)
  64001  |************************************* (383)
  65001  |****************************** (308)
  66001  |******************************************************************************** (812)
  67001  |********************************************************************** (712)
  68001  |******************************************** (455)
  69001  |********************************************************** (598)
  70001  |************************************** (391)
  71001  |**************************************** (413)
  72001  |******************************************************** (573)
  73001  |************************************** (386)
  74001  |****************************************************** (554)
  75001  |********************************************************** (597)
  76001  |************************************* (377)
  77001  |******************************************* (440)
  78001  |********************************* (338)
  79001  |********************************* (338)
  80001  |*********************************** (357)
  81001  |****************************** (308)
  82001  |************************************ (370)
  83001  |********************************* (339)
  84001  |************************************************************************** (758)
  85001  |********************************************************************** (718)
  86001  |***************************************************** (540)
  87001  |****************************** (313)
  88001  |************************* (254)
  89001  |****************************** (308)
  90001  |*************************************** (399)
  91001  |********************************************* (459)
  92001  |*********************************** (357)
  93001  |******************************************** (448)
  94001  |************************************* (383)
  95001  |***************************** (301)
  96001  |*************************** (275)
  97001  |**************************** (285)
  98001  |******************************* (319)
  99001  |************************************** (392)
```

<u>Algorithm 4:</u>
Algorithm four has a time complexity of O(n). This is because algorithm 4 has one for loop that iterates through the vector “lis_arr” in order to find the “target” number (another searching algorithm) and since there are n elements in the vector, the runtime complexity would have to be O(n). The graph of algorithm 4 also has a more linear growth and is similar to the graph of the function n.

Algorithm 4 Graph -

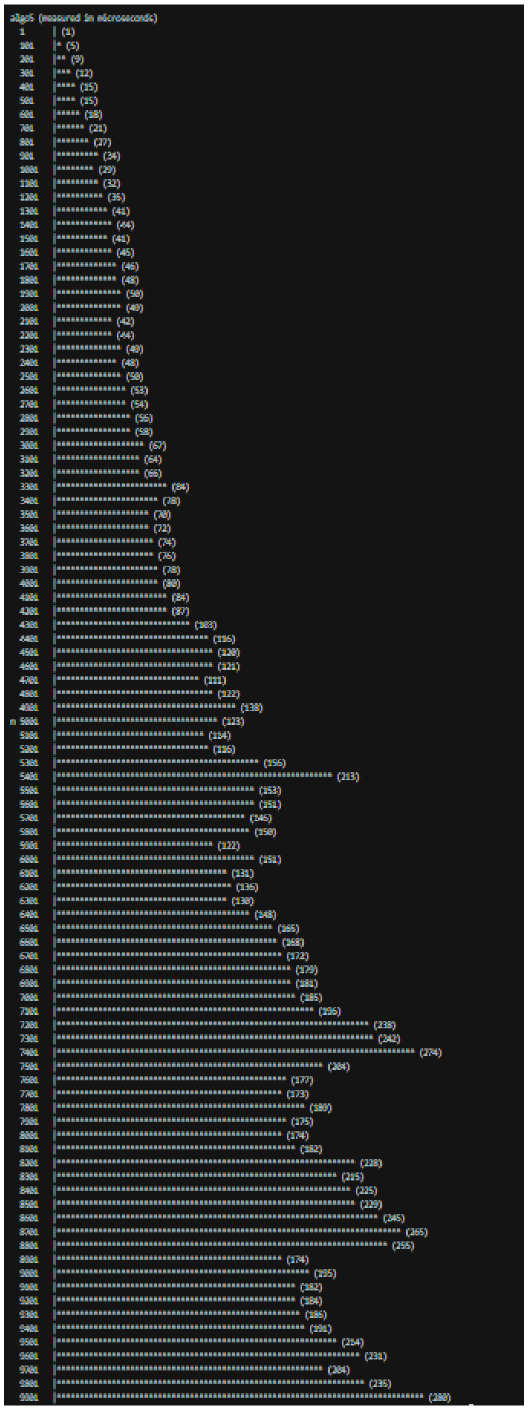


```
algo4 (measured in microseconds)
  1      | (0)
  101    | (0)
  201    | (0)
  301    | (0)
  401    |* (1)
  501    |* (1)
  601    |* (1)
  701    |* (1)
  801    |** (2)
  901    |** (2)
  1001   |** (2)
  1101   |** (2)
  1201   |** (2)
  1301   |*** (3)
  1401   |*** (3)
  1501   |*** (3)
  1601   |*** (3)
  1701   |***** (4)
  1801   |***** (4)
  1901   |***** (4)
  2001   |***** (4)
  2101   |***** (4)
  2201   |****** (5)
  2301   |****** (5)
  2401   |****** (5)
  2501   |****** (5)
  2601   |******* (6)
  2701   |******* (6)
  2801   |******* (6)
  2901   |******* (6)
  3001   |******** (7)
  3101   |******** (7)
  3201   |******** (7)
  3301   |******** (7)
  3401   |********** (8)
  3501   |********** (8)
  3601   |********** (8)
  3701   |********** (8)
  3801   |*********** (9)
  3901   |********** (8)
  4001   |*********** (9)
  4101   |*********** (9)
  4201   |*********** (9)
  4301   |*********** (9)
  4401   |************ (10)
  4501   |************ (10)
  4601   |************ (10)
  4701   |************ (10)
  4801   |************* (11)
  4901   |************* (11)
n 5001   |************* (11)
  5101   |************* (11)
  5201   |************* (11)
  5301   |*************** (12)
  5401   |*************** (12)
  5501   |*************** (12)
  5601   |*************** (12)
  5701   |**************** (13)
  5801   |**************** (13)
  5901   |********************* (17)
  6001   |********************** (18)
  6101   |********************* (17)
  6201   |**************** (13)
  6301   |***************** (14)
  6401   |******************************************************************************** (64)
  6501   |************************** (21)
  6601   |*********************** (19)
  6701   |************************** (21)
  6801   |******************************* (25)
  6901   |********************************* (27)
  7001   |******************************** (26)
  7101   |****************************** (24)
  7201   |********************* (17)
  7301   |********************* (17)
  7401   |********************* (17)
  7501   |********************* (17)
  7601   |********************* (17)
  7701   |********************** (18)
  7801   |************************* (20)
  7901   |************************** (21)
  8001   |************************** (21)
  8101   |************************* (20)
  8201   |**************************** (23)
  8301   |****************************** (24)
  8401   |************************** (21)
  8501   |*********************** (19)
  8601   |**************************** (23)
  8701   |*************************** (22)
  8801   |************************* (20)
  8901   |************************** (21)
  9001   |**************************** (23)
  9101   |****************************** (24)
  9201   |******************************** (26)
  9301   |********************************* (27)
  9401   |********************************* (27)
  9501   |********************************* (27)
  9601   |******************************* (25)
  9701   |************************************** (31)
  9801   |************************************** (31)
  9901   |******************************************* (35)
```

<u>Algorithm 5:</u>

Algorithm five has a time complexity of O(n). This is because there is a single loop in order to add data into the vector “powers”. This loop also calls the helper function within the loop which uses recursion in order to find the value of “power” for each index and this helper function has a complexity of O(logn), but since n is always 11 and log(11) is a constant, this can be ignored. Hence this would mean that the overall complexity would be O(n). The graph of algorithm 5 also has a steady growth which indicates that the time complexity is indeed O(n).

Algorithm 5 Graph -



```
algo5 (measured in microseconds)
  1       | (1)
  101     |* (5)
  201     |** (9)
  301     |*** (12)
  401     |**** (15)
  501     |**** (15)
  601     |***** (18)
  701     |****** (21)
  801     |******* (27)
  901     |********* (34)
  1001    |******** (29)
  1101    |********* (32)
  1201    |********** (35)
  1301    |*********** (41)
  1401    |************ (44)
  1501    |*********** (41)
  1601    |************ (45)
  1701    |************* (46)
  1801    |************* (48)
  1901    |************** (50)
  2001    |************** (49)
  2101    |************ (42)
  2201    |************ (44)
  2301    |************** (49)
  2401    |************* (48)
  2501    |************** (50)
  2601    |*************** (53)
  2701    |*************** (54)
  2801    |**************** (56)
  2901    |**************** (58)
  3001    |******************* (67)
  3101    |****************** (64)
  3201    |****************** (66)
  3301    |************************ (84)
  3401    |********************** (78)
  3501    |******************** (70)
  3601    |******************** (72)
  3701    |********************* (74)
  3801    |********************* (76)
  3901    |********************** (78)
  4001    |********************** (80)
  4101    |************************ (84)
  4201    |************************ (87)
  4301    |***************************** (103)
  4401    |********************************* (116)
  4501    |********************************** (120)
  4601    |********************************** (121)
  4701    |******************************* (111)
  4801    |********************************** (122)
  4901    |*************************************** (138)
n 5001    |*********************************** (123)
  5101    |******************************** (114)
  5201    |********************************* (116)
  5301    |******************************************** (156)
  5401    |************************************************************ (213)
  5501    |******************************************* (153)
  5601    |******************************************* (151)
  5701    |***************************************** (146)
  5801    |****************************************** (150)
  5901    |********************************** (122)
  6001    |******************************************* (151)
  6101    |************************************* (131)
  6201    |************************************** (136)
  6301    |************************************* (130)
  6401    |****************************************** (148)
  6501    |*********************************************** (165)
  6601    |************************************************ (168)
  6701    |************************************************* (172)
  6801    |*************************************************** (179)
  6901    |*************************************************** (181)
  7001    |**************************************************** (185)
  7101    |******************************************************** (196)
  7201    |******************************************************************** (238)
  7301    |********************************************************************* (242)
  7401    |****************************************************************************** (274)
  7501    |********************************************************** (204)
  7601    |************************************************** (177)
  7701    |************************************************* (173)
  7801    |*************************************************** (180)
  7901    |************************************************** (175)
  8001    |************************************************* (174)
  8101    |**************************************************** (182)
  8201    |***************************************************************** (228)
  8301    |************************************************************* (215)
  8401    |**************************************************************** (225)
  8501    |***************************************************************** (229)
  8601    |********************************************************************** (245)
  8701    |*************************************************************************** (265)
  8801    |************************************************************************ (255)
  8901    |************************************************* (174)
  9001    |******************************************************* (195)
  9101    |**************************************************** (182)
  9201    |**************************************************** (184)
  9301    |***************************************************** (186)
  9401    |****************************************************** (191)
  9501    |************************************************************* (214)
  9601    |****************************************************************** (231)
  9701    |********************************************************** (204)
  9801    |******************************************************************* (235)
  9901    |********************************************************************************** (288)
```