# Investigate the binary search algorithm

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | def binary\_search(items,target):    low=0  high=len(items)-1  mid=high//2  while target != items[mid]:  mid=(low+high)//2  if items[mid] < target:  low=mid+1  elif items[mid] > target:  high=mid-1    return items[mid], mid  items=[0,5,13,19,22,41,55,68,72,81,98]  target=33  print(binary\_search(items,target)) |

|  |  |
| --- | --- |
| 1. Identify the variables | low, high, mid, target |
| 1. What is are the initial values of low, high and mid? | 0, 10, 5 |
| 1. What is the // operator doing? | Integer division |
| 1. What are the parameters to the function binary\_search | Items, and target |
| 1. How many parameters does the function binary\_search take | 2 |
| 1. Give the example of selection in the code | Lines 7 and 9 |
| 1. Give the example of iteration | While loop on line 5 |
| 1. Under what condition does the loop stop | When the target item has been found |
| 1. What do you notice about the values in the items list? | They are in ascending order |
| 1. Explain what is happening in lines 7-10 | Determines which half of the list to keep by determining if the value mid value is bigger or smaller than the target value |
| 1. Overall what is the code doing? | Halving the search area at each iteration in order to converge on the index position of the target value. |