Jason Pototsky

CST 200

1/28/2014

Given a Java String array containing the following values:

apple	cherry	grapefruit	lemon	orange	pear	strawberry
0	1	2	3	4	5	6

a) How many probes of the binary search are required to locate *lemon*?
Find mid of array Min 1 Max 7 1+7= 8/2 = 4-1 = 3 Mid
1. Goto Mid of Index 3 which is lemon > lemon::no (Eliminate 4/5/6)
2. Get new adjusted mid Min 0 Max 3+1 =4 /2 = 2-1 = 1 Mid
3. Goto Mid of Index 3 which is lemon > lemon::no(Eliminate 0/1)
4. Get new adjusted mid Min 3 Max 4+1 =8/2 = 4 -1 = 3 Mid
5. Goto Mid of Index 3 is lemon < lemon::no(Eliminate 3)
6. Only 1 cell remains check is lemon == lemon::yes
5. Return index 3

Probes without equality check is 4
Probes with equality check is 1 or 2 depending on if your equality check comes before the check for greater than.

b) How many probes of the binary search are required to locate *apple*?
c) Find mid of array Min 1 Max 7 1+7= 8/2 = 4-1 = 3 Mid
1. Goto Mid Index 3 is lemon < apple::no (Eliminate index 4/5/6)
2. Get new adjusted mid Min 1 Max 4+1 =6 /2 = 3-1 = 2 Mid
3. Goto Mid of Index 2 which is grapefruit > apple::yes(Eliminate 2/3)
4. Get new adjusted mid Min 1 Max 2+1 = 4/2 = 2-1 = 1 Mid
5. Goto Mid of Index 1 which is cherry > apple::yes(Eliminate index 1)
6. One value remains checks is apple == apple::yes
7. Return index 0

Probes without equality check 4
Probes with equality check 7

d) How many probes of the binary search are required to locate *orange*?
Find mid of array Min 1 Max 7 1+7= 8/2 = 4-1 = 3 Mid
1. Goto Mid Index 3 is lemon < orange::yes (Eliminate index 0/1/2/3)
2. Get new adjusted Mid Min 4 Max 6+1 = 11 /2 = 5.5~6-1= 5 Mid
3. Goto Mid Index 5 is pear < orange::no (Eliminate index 6)

4. Get new adjusted Mid Min 4 Max 5+1 = 10/2 = 5-1 = 4 Mid
5. Goto Mid Index 4 is orange > orange::no (Eliminate index 5)
6. Only 1 Cell remains is orange == orange::yes
7. Return index 4

Probes without equality check is 4
Probes with equality check is 7

e) How many probes of the binary search are required to determine *melon* is not in the list?

Find mid of array Min 1 Max 7 1+7= 8/2 = 4-1 = 3 Mid
1. Goto Mid Index 3 is lemon < melon::yes (Eliminate index 0/1/2/3)
2. Get new adjusted Mid min 4 Max 6+1=11/2=5.5~6-1= 5 Mid
3. Goto Mid Index 5 is pear < melon::no (Eliminate 4)
4. Get new adjusted Mid min 5 Mac 6+1= 12/2 = 6-1 = 5 Mid
5. Goto Mid Index 5 is pear >melon::yes (Eliminate 6)
6. Only 1 Cell remains is pear == melon::no
7. Return index -1

Probes without equality check 4
Probes with equality check is 7

Given a Java String array containing the following values:

apple	cherry	grapefruit	lemon	orange	pear	strawberry

f) How many probes of the sequential/linear search are required to locate *lemon*?

4 Probes.

1. Is Index 0 apple == lemon::no
2. Is Index 1 cherry == lemon::no
3. Is Index 2 grapefruit == lemon::no
4. Is Index 3 lemon == lemon::yes
5. Return Index 3

g) How many probes of the sequential/linear search are required to locate *apple*?

1 Probe

1. Is Index 0 apple = apple::yes
2. Return Index 0

h) How many probes of the sequential/linear search are required to locate *orange*?

5 Probes

1. Is Index 0 apple == orange::no
2. Is Index 1 cherry == orange::no
3. Is Index 2 grapefruit == orange::no
4. Is Index 3 lemon == orange::no
5. Is Index 4 orange == orange:yes
6. Return Index 4

i) How many probes of the sequential/linear search are required to determine *melon* is not in the list?

7 Probes

1. Is Index 0 apple == melon::no
2. Is Index 1 cherry == melon::no
3. Is Index 2 grapefruit == melon::no
4. Is Index 3 lemon == melon::no
5. Is Index 4 orange == melon:no
6. Is Index 5 pear == melon:no
7. Is Index 6 strawberry == melon:no
8. Return Index -1