

Cmp 338

①

9.24.19

Gate Keeper



public class StackObj

private int top = -1

private Object[] A;

public StackObj(int size)

{

A = new Object[size];

}

public void push(Object v)

{

Change int to
Object

}

public

etc

Changing int or
string or
char

to Object

Gate Keeper

public class StackIntObj

{

StackObj stack;

public StackIntObj(int size)

{

stack = new StackObj(size);

}

public void push(int x)

{

stack.push(new Integer(x));

{

public int top() ← pop() is String

{

return (Integer)(stack.top(), Integer);

}

public boolean isEmpty()

{

return stack.isEmpty();

}

Cmp 338

②

9.24.09



LIST

LIST ADT

Insert
Remove
Search

AS long as the list is sorted

$$m = (F+L)/2$$

\log_2

2047
1023
511
255
127
63
31
15
7
3
1

$$\log_{10} 10^9 = \log_{10} (10^3 \cdot 10^3 \cdot 10^3) = 30$$

X
56

0 1 2 3 4 etc
□ □ □ □ □

$$\begin{aligned} \log_{10} 10^6 &= \log_{10} (10^3 \cdot 10^3) \\ &= \log_{10} 10^3 + \log_{10} 10^3 \approx \log_2 2^{10} + \log_2 2^{10} \\ &= 10 + 10 = 20 \end{aligned}$$

$\log_2 n - \log_2 n$ - report the number of times n can be divided by 2 before reaching one.

$$\log_{10} 10000 = 4$$

$$\log_{10} 10^7 \approx \log_{10} 2^{21}$$

$$\log(a \cdot b) = \log a + \log b$$

Cmp 338

③

9.24.09

public int binarySearch (int[] A, int first, int last, int x)
{

int middle = first + last/2;

if (x == A[middle]) { return middle; }

if (x < A[middle]) { return binarySearch(x, A, first, middle-1); }

if (x > A[middle]) { return binarySearch(x, A, middle+1, last); }