CS100J 01 November 2007

Arrays: searching & sorting. Reading: 8.5

Searching and sorting algorithms are on the course website.

Why did I get a Christmas Card on Halloween?

Decimal	Octal	Binary	Decimal	Octal	Binary	Decimal	Octal	Binary
00	00	0000	11	13	01011	22	26	010110
01	01	0001	12	14	01100	23	27	010111
02	02	0010	13	15	01101	24	30	011000
03	03	0011	14	16	01110	25	31	011001
04	04	0100	15	17	01111	26	32	011010
05	05	0101	16	20	10000	27	33	011011
06	06	0110	17	21	10001	28	34	011100
07	07	0111	18	22	10010	29	35	011101
08	10	1000	19	23	10011	30	36	011110
09	11	1001	20	24	10100	31	37	011111
10	12	1010	21	25	10101	32	40	100000

1

Decimal 5482: $5*10^3 + 4*10^2 + 8*10^1 + 2*10^0$ = 5482 in decimal

Octal 3726: $3*8^3 + 7*8^2 + 2*8^1 + 6*10^0$ = 2006 in decimal

Binary 1011: $1*2^3 + 0*2^2 + 1*2^1 + 1*2^0$ = 11 in decimal

```
/** = a string that contains the binary representation of n.
      Precondition: n >= 0 */
public static String binary(int n) {
  if (n <= 1)
     return "" + n;

  return  binary(n/2)  +  (n%2);
}
```

2

Developing algorithms on arrays

The rest of this lecture develops several important algorithms on arrays.

With each, we specify the algorithm by giving its precondition and postcondition as pictures.

Generally, the invariant comes from drawing another picture that "generalizes" the precondition and postcondition, since the invariant is true at the beginning and at the end.

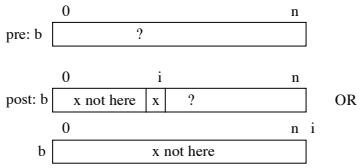
Four loopy questions —memorize them:

1. How does loop start (how to make the invariant true)?
2. When does it stop (when is the postcondition true)?
3. How does repetend make progress toward termination?
4. How does repetend keep the invariant true?

3

Getting an invariant as picture:

• Linear search. Vague spec.: find first occurrence of v in b[h..k-1].
Better spec.: Store an integer in i to truthify:
postcondition: (0) v is not in b[h..i-1]
(1) Either i= k or v = b[k]

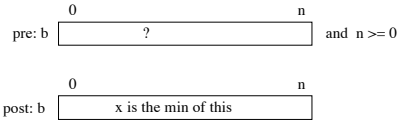


4

Getting an invariant as picture:

Combine pre- and post-condition

Finding the minimum of an array

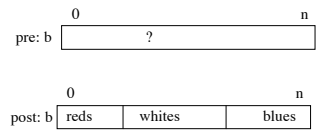


5

Getting an invariant as picture:

Combine pre- and post-condition

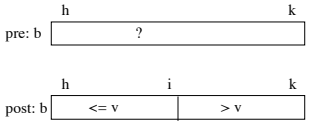
Dutch national flag. Array



6

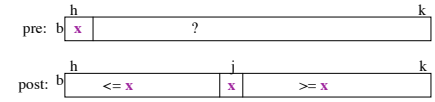
Binary search: Vague spec: Look for v in sorted array segment b[h..k].
Better spec:
Precondition: b[h..k] is sorted (in ascending order).
Store in i to truthify:
postcondition: b[h..i] <= v and v < b[i+1..k]

Below, the array is in non-descending order



Partition algorithm:

x is called the pivot value



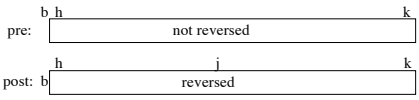
change: b 3 5 4 1 6 2 3 8 1 (h, k)

into b 1 2 1 3 5 4 6 3 8 (h, j, k)

or b 1 2 3 1 3 4 5 6 8 (h, j, k)

(x is not a program variable; it just denotes the value initially in b[h].)

Reversing array segment b[h..k]



change: b 1 2 3 4 5 6 7 8 9 9 9 9 (h, k)

into b 9 9 9 9 8 7 6 5 4 3 2 1 (h, j, k)

Remove adjacent duplicates

change: b 1 2 2 4 2 2 7 8 9 9 9 9 (0, n)

into b 1 2 4 2 7 8 9 8 9 9 9 9 (0, h, n)

don't care what is in b[k+1..n]

postcondition:

b[0..h] = initial values in b[0..n] but with adj dups removed

b