

REC3: loop-invariant problems

1.

- (1) Does it start right? Is $\{Q\}$ init $\{P\}$ true?
- (2) Does it end right? Is $P \ \&\& \ !B \Rightarrow R$ true?
- (3) Does the repetend make progress toward termination?
- (4) Does the repetend maintain P: Is $\{P \ \&\& \ B\} \ S \ \{P\}$ true?

2.

- 2A) To make $c[0..k-1]$ empty, let $k - 1 = -1$, so init: **int k = 0;**
- 2B) To make $c[k..c.length-1]$ empty, let $c.length - 1 = k - 1$, so init: **int k = c.length;**
- 2C) To make $c[k+1..c.length-1]$ empty, let $k + 1 = c.length - 1 + 1$, so init: **int k = c.length - 1;** thus s would be equal to 1

3.

- 3A) !B is $k = c.length$, so B: $k \neq c.length$
- 3B) !B is $k = 0$, so B: $\text{int } k \neq 0$;
- 3C) !B is $k + 1 = 0$ so B: $\text{int } k \neq 1$;

4.

- 4A)


```

      if (c[k] != 0)
          b = false; // should I write break; ?
      k = k + 1;
      
```
- 4B)


```

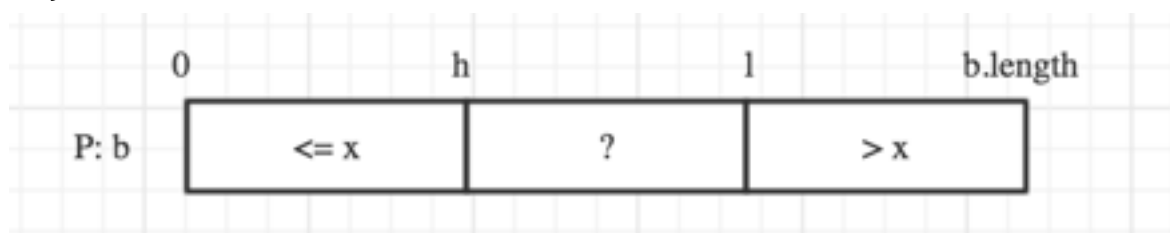
      if (c[k-1] != 0)
          b = false;
      k = k - 1;
      
```
- 4C)


```

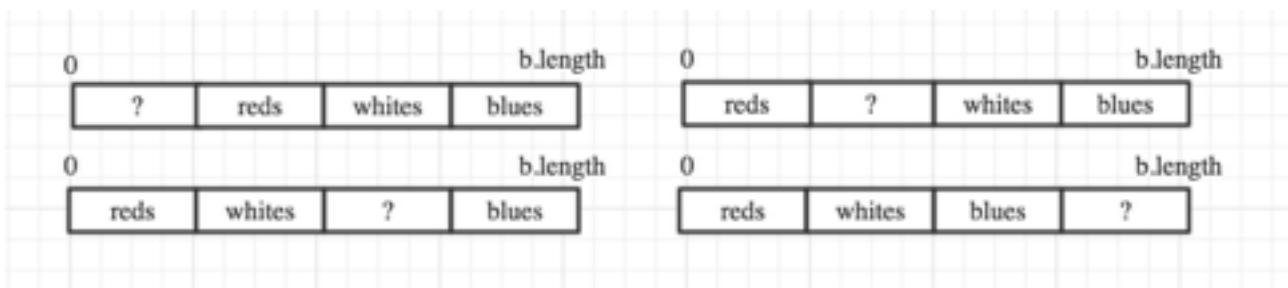
      s = s*c[k];
      k = k - 1;
      
```

5.

5A)



5B)



6.

```
h = b.length - 1;
while (0 <= h && b[h] != v)
    h = h - 1;
```

7.

```
h = -1;
i = b.length;
while (h + 1 < i){
    if (b[h + 1] <= x)
        h = h + 1;
    else
        i = h + 1;
}
```

8.

```
j = h;
i = k;
while (j < i){
    if (b[j + 1] <= b[h])
        j = j + 1;
    else{
        temp = b[j + 1];
        b[j + 1] = b[i];
        b[i] = temp;
        i = i - 1;
    }
}
```