

## Homework 2, week 4

**1. Bitcount.** In the textbook (page 50), there is a function `bitcount` counting the number of 1-bits in its integer argument.

```
1. /* bit count: count the number of 1-bits in x */
2. int bitcount (unsigned x) {
3.     int b;
4.     for (b = 0; x!=0; x>>= 1) {
5.         if (x & 01) b++;
6.     }
7.     return b;
8. }
```

you can improve this function by using a different method using the observation that `x &=(x-1)` will delete the rightmost 1-bit in `x`. Please explain why `x &= (x -1)` will delete the right most 1-bit of `x`, and write a faster version of `bitcount`.

**2. Rewrite the following function `lower`, which converts upper case letter to lower case. Please use a conditional expression(?:) instead of if-else.**

```
1. /* lower: convert upper case letter to lower case */
2. int lower (int c) {
3.     if (c >= 'A' && c <= 'Z')
4.         return c + 'a' - 'A';
5.     else
6.         return c;
7. }
```

**3. Write a function `any(s1, s2)`, which returns the first location in the string `s1` where any character from the string `s2` occurs, or -1 if `s1` contains no characters from `s2`. You can use the `squeeze()` function listed here as your reference.**

```
1. /* squeeze: delete all c from s, return the number of deleted letters*/
2. int squeeze (char s[], int c) {
3.     int i, j, n = 0;
4.     for (i = j = 0; s[i] != '\0'; i++)
5.         if (s[i] != c) {
```

6. `s[j++] = s[i];`

7. `n++;`

8. `}`

9. `S[j] = '\0';`

10. `return n;`

11. `}`

**You need to modify, save, compile, test the code and then report all details including the test results for each program.**

**Turning in your homework**

Please hand in a hard copy of your homework report. The homework report should be handed in before the class start on April 1st, 2014.

-----cut----here-----

独立作业承诺：（请选择一个，并签名）

1. 本人， ，保证本次作业由自己独立完成。

签名

时间 年 月 日

或者

2. 本人， ，保证本次作为和 同学讨论后，由自己独立完成。

讨论内容包括

签名，

时间 年 月 日