Hw 1

Page12: 2.1-2, 2.1-4

Page16: 2.2-2, 2.2-3

2.1

2.1-2 重写过程 INSERTION-SORT，使之按非升序(而不是非降序)排序。

书上非降序排序:

```
INSERTION-SORT(A)
1  for j = 2 to A.length
2      key = A[j]
3      // Insert A[j] into the sorted sequence A[1..j - 1].
4      i = j - 1
5      while i > 0 and A[i] > key
6          A[i+1] = A[i]
7          i = i - 1
8      A[i+1] = key
```

非升序排序：只要对上面第5行修改：A[i] < key

2.1-4 考虑把两个 n 位二进制整数加起来的问题，这两个整数分别存储在两个 n 元数组 A 和 B 中。这两个整数的和应按二进制形式存储在一个 $(n+1)$ 元数组 C 中。请给出该问题的形式化描述，并写出伪代码。

输入：存储两个n位二进制数的n元数组A[1..n], B[1..n]

输出：存储A, B对应二进制数的和的n+1元数组C[1..n+1]

伪码:

```
BinaryAdd(A, B):
# 输入：A[1..n], B[1..n]。从数组最高位A[n], B[n]开始存二进制数。
# 输出：C[1..n+1]。也是从最高位C[n+1]开始存输入两个二进制数的和。
    # carry表进位
    carry = 0
    # 二进制从数组的最高位开始存，所以i=n to 1，循环体内用C[i+1]
    for(i=n to 1):
        sum = A[i]+B[i]+carry
        #  应该是C[i+1]，不是C[i]
        C[i+1] = sum % 2
        carry = floor(sum/2)
    # 最后要将C[1]置为carry
    C[1] = carry
```

- 如果用C[i]存carry，最后就不用C[1]=carry
- 也可以将二进制数倒着存，即从数组最低位开始存，其实这样更简单

法2：如果if-else分类讨论，则要考虑sum=0, 1, 2, 3的情况。

2.2

2.2-2 考虑排序存储在数组 A 中的 n 个数：首先找出 A 中的最小元素并将其与 $A[1]$ 中的元素进行交换。接着，找出 A 中的次最小元素并将其与 $A[2]$ 中的元素进行交换。对 A 中前 $n-1$ 个元素按该方式继续。该算法称为**选择算法**，写出其伪代码。该算法维持的循环不变式是什么？为什么它只需要对前 $n-1$ 个元素，而不是对所有 n 个元素运行？用 Θ 记号给出选择排序的最好情况与最坏情况运行时间。

升序，选择排序

1. 伪码：

```
SelectionSort(A):
# 输入：A[1..n]
# 输出：升序排好的A[1..n]
    n = len(A)
    # 最后一个元素不用排，所以是i to n-1
    for i = 1 to n-1:
        minId = i
        # 找i之后的元素更新minId，所以是从i+1开始
        for j = i+1 to n:
            if A[j] < A[minId]:
                minId = j
        swap(A, i, minId)
```

2. 循环不变式（证明算法正确性）：
    1. 初始化
    2. 保持：排序好的部分保持有序
    3. 终止：能正确中止
3. 因为最后一个元素一定是最大的
4. 最好最坏都要依次在n-1, n-2,..,1个元素中找最小元素，然后交换，时间$\Sigma_{i=1}^{n-1} i = n(n-1)/2$

   $\theta(n^2)$

2.2-3 再次考虑线性查找问题(参见练习 2.1-3)。假定要查找的元素等可能地为数组中的任意元素，平均需要检查输入序列的多少元素？最坏情况又如何呢？用 Θ 记号给出线性查找的平均情况和最坏情况运行时间。证明你的答案。

2.1-3 考虑以下**查找问题**：

输入：n 个数的一个序列 $A=\langle a_1, a_2, \cdots, a_n\rangle$和一个值 v。

输出：下标 i 使得 $v=A[i]$或者当 v 不在 A 中出现时，v 为特殊值 NIL。

写出**线性查找**的伪代码，它扫描整个序列来查找 v。使用一个循环不变式来证明你的算法是正确的。确保你的循环不变式满足三条必要的性质。

1. 平均需要检查：

   v出现的位置求个期望：n个元素每个出现的概率都是$1/n$，故期望位置$\Sigma_{i=1}^{n} i/n = (n+1)/2$

   最坏需要检查n个元素

2. 平均和最坏都是 $\theta(n)$。证明：由 *1* ，平均和最坏都要检查$\theta(n)$量级的元素，一次检查耗时$\theta(1)$.

9月24日随堂测试

已知定理：

$f(x)$ 是任何连续单调上升函数，且 $f(x)$ 在整数点才可能取整数值，则：

1) $$\lfloor f(x) \rfloor = \lfloor f(\lfloor x \rfloor) \rfloor$$

2) $$\lceil f(x) \rceil = \lceil f(\lceil x \rceil) \rceil$$

证明：

$$\left\lceil \frac{\lceil n/a \rceil}{b} \right\rceil = \left\lceil \frac{n}{ab} \right\rceil \qquad \left\lfloor \frac{\lfloor n/a \rfloor}{b} \right\rfloor = \left\lfloor \frac{n}{ab} \right\rfloor$$

（二者证明一个即可）

取$f(x) = x/b$，$x = n/a$即可。