Q1.（15 分）教材练习 7.2.6

Q2.（35 分）

(1) 给定有限数列 $a_1, a_2, \dots, a_n$ 以及有限正项数列 $w_1, w_2, \dots, w_n$。定义 $\{a_n\}$ 以 $\{w_n\}$ 为权、以 x 为中心的加权平均差：

$$D(x) = \frac{\sum_{i=1}^{n} w_i |a_i - x|}{\sum_{i=1}^{n} w_i}$$

试设计一个具有 $O(n)$ 最坏时间复杂度的算法求出 $\min\limits_{x} D(x)$

(*2) 对于数列 $\{a_n\}$，定义：

$$\Delta\{a_n\} = \Delta^1\{a_n\} = \{a_{n+1} - a_n\}$$
$$\Delta^k\{a_n\} = \left(\Delta \circ \Delta^{k-1}\right)\{a_n\}, k \geq 2$$

称 Δ 为差分算子，$\Delta^k\{a_n\}$ 为 $\{a_n\}$ 的 k 阶差分数列。

若 T 同时是纯周期数列 $\{a_n\}$ 和 $\{b_n\}$ 的周期，且 $\Delta\{a_n\} = \Delta^2\{b_n\}$。给定 $a_1, a_2, \dots, a_T$，试设计一个具有 $O(T)$ 最坏时间复杂度的算法求出 $\sum\limits_{1 \leq i \leq T} |b_i|$ 的最小可能值。

Q3.（$10 + 20 = 30$ 分）

(1) 给定两个严格升序数组 A, B，长度分别为 n, m，且不存在 i, j 使得 $A[i] = B[j]$。定义 $S = \{\langle i, j\rangle \mid A[i] > B[j]\}$，请设计一个线性最坏时间复杂度的算法求出 $|S|$

(2) 给定长度为 n 的数组 L，数组内无重复元素。定义 $I = \{\langle i, j\rangle \mid i < j \wedge L[i] > L[j]\}$，请设计一个最坏时间复杂度为 $O(n \lg n)$ 的算法求出 $|I|$

(*3) 对于 $C \subseteq \{1, 2, \dots, n\}^2$ 以及长度为 n 的不含重复元素的数列 A，如果 $\forall \langle i, j\rangle \in C, A[i] < A[j]$，则称 "$A$ 服从于 C"。

证明：若存在 A 的两个不同的排列均服从于 C，那么进一步地，一定存在 A 的两个排列 A_1, A_2，它们都服从于 C，且只相差一次对换，即存在 $i \neq j$，使得：

$$\begin{cases} A_1[i] = A_2[j] \\ A_1[j] = A_2[i] \\ A_1[k] = A_2[k], \forall k \neq i, j \end{cases}$$

(*4) 利用 (3) 中得到的引理证明：对于 (2) 中的问题，基于比较的算法具有 $\Omega(n \lg n)$ 的最坏时间复杂度下界。

Q4.（$10+10=20$ 分）打乱算法用于随机地将序列中的元素进行重排。原则上，打乱算法不能利用任何元素本身的信息，例如不能进行两个元素间的比较。

算法1、2、3、4分别是四种打乱算法的实现，假设输入数列为 L，长度为 n，且 L 中无重复元素。算法中`randint(a,b)` 表示在 $[a,b]$ 中随机选择整数。`swap(i,j)` 表示交换 $L[i],L[j]$ 的值，如果 $i=j$ 则不进行操作。

(1) 若某打乱算法可以生成所有可能的元素排列，则称其为“完备”的。这些算法是完备的吗？如果是，请证明；如果不是，请举出反例。

(2) 若某打乱算法以等概率生成所有它能够生成的元素排列，则称其为“均匀”的。当 $n\geq 3$ 时，这些算法是均匀的吗？如果是，请证明；如果不是，请说明理由。

(*3) 以下给出了两个断言，判断它们是否正确，并简要说明理由。(假设 $n\geq 3$)

(*3.1) 若某打乱算法是完备的，且任一元素被打乱到每一个位置的概率都为 $\frac{1}{n}$，则它一定是均匀的。

(*3.2) 若某打乱算法是完备的，无论它是否均匀，只要将其重复执行多次，最终的结果一定趋于均匀。

(*4) 将谭教授安排的任务处理结束后，苏雨沫小姐回到家中与一名杰出的博士玩起了扑克牌游戏，规则如下：博士拿出点数及初始顺序为“J、Q、K”的三张扑克牌，然后将其打乱。苏小姐对打乱后的顺序进行多次猜测，直到猜对为止（显然，至多 6 次猜测就可以结束一轮）。几轮游戏后，苏小姐发现自己能够通过观察手部动作确定博士本轮采用了哪种打乱方式，但她不知道如何利用这一信息，于是她立刻向你求助，希望你帮忙设计最优策略以最小化猜测次数的期望值。作为苏小姐最宠爱的小师妹（弟），你当然不会辜负大师姐的期望，因此请分别显式地给出四种打乱方式下的最优策略，并分别计算出猜测次数期望值。

(*5) 若某打乱算法恰能生成所有的错位排列，即任何一个元素都不会停留在原地，则称其为“错位”的。假设算法中每次随机数获取都需要从真随机源中抽取相应大小的熵。试证明：均匀错位打乱算法的期望熵使用量的下界是 $\Omega(n\lg n)$，并设计一个达到此下界的算法。

Algorithm 1: Shuffle-1

```
for i ← 1 to n do
    j ← randint(1, n);
    swap(i, j);
end
```

Algorithm 2: Shuffle-2

```
for i ← 1 to n do
    for j ← 1 to n do
        if randint(0, 1) = 1 then
            swap(i, j);
        end
    end
end
```

Algorithm 3: Shuffle-3

```
for k ← 1 to n do
    i ← randint(1, n);
    j ← randint(1, n);
    swap(i, j);
end
```

Algorithm 4: Shuffle-4

```
for i ← 1 to n do
    j ← randint(i, n);
    swap(i, j);
end
```

***Q5.** 给定有限集 X，在其上定义了全预序“$\lesssim$”。记 $\lesssim$ 导出的严格偏序和等价关系分别为“$<$”、“$\sim$”。

集合 X^m 上的字典序“$\lesssim_m$”定义为：

$$
\begin{aligned}
&\langle x_1, x_2, \ldots, x_m \rangle \lesssim_m \langle y_1, y_2, \ldots, y_m \rangle \\
&\iff \left(\bigwedge_{i=1}^{m} x_i \sim y_i \right) \vee \left(\bigvee_{i=0}^{m-1} \left((x_{i+1} < y_{i+1}) \wedge \left(\bigwedge_{j=1}^{i} x_j \sim y_j \right) \right) \right)
\end{aligned}
$$

(*1) 证明 $\lesssim_m$ 是 X^m 上的全预序。

以下记 $\lesssim_m$ 导出的严格偏序和等价关系分别为“$<_m$”和“$\sim_m$”。

(*2) 假设 $X = \{a, b, c, d, e\}$

(*2.1)X 上不同的全预序共有多少个？其中同构意义下不同的全预序共有多少个？两个预序 $\lesssim$ 与 $\lesssim'$ 同构是指：存在 X 上的置换 φ，使得 $x \lesssim y \iff \varphi(x) \lesssim' \varphi(y)$

(*2.2) 请任选一个全预序，以哈斯图的形式表示 $\lesssim$ 和 $\lesssim_2$ 导出的偏序，并分别写出商集 $X/\sim$ 和 $X^2/\sim_2$

(*3) 对于 $x \in X^m$，记 $\mathbb{L}(x) = \{y \mid y <_m x\}, \mathbb{LS}(x) = \{y \mid y \lesssim_m x\}$。称 x 是“第 k 小的”，当且仅当 $|\mathbb{L}(x)| < k \leq |\mathbb{LS}(x)|$。试设计算法，以 $X, \lesssim, m, k$ 作为输入，返回 X^m 中第 k 小的元素（若有多个，只需返回任意一个），要求最坏时间复杂度：

$$
T_{\text{worst}}(X, m) = \max_{\lesssim, k} T(X, \lesssim, m, k) = O(m + |X| \lg |X|)
$$

假设 X 中任意两个元素在 $\lesssim$ 意义下的比较均只需要常数时间。