2020 年春季学期课程期末考试试卷答题纸

课程名称：算法设计与分析 **课程代码：COMP130011.02**

卷别：■A 卷□B 卷□C 卷

姓名： **学号：**

我已知悉学校与考试相关的纪律以及违反纪律的后果，并将严守纪律，不作弊，不抄袭，独立答题。

学生（签名）：

年 月 日

题号	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**总分**
得分								

1. 分治法 (10 分)

在 Ailon 等的 FJLT 算法中应用了 Hadamard 矩阵的一个性质，其中 Hadamard 矩阵 H_k 为 $2^k \times 2^k$ 的矩阵，定义如下：

$$H_1 = [1],\ H_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \ldots,\ H_k = \frac{1}{\sqrt{2}}\begin{bmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & -H_{k-1} \end{bmatrix}$$

记 $n = 2^k$，给定一个 n 维列向量 $x \in R^n$, 设计一个时间复杂度为 $O(n \log n)$的算法来完成矩阵 H_k 和向量 x 的乘积 $H_k x$.

2. Hash函数应用 (10分)

我们需对一条数据流$(i_1, c_1), (i_2, c_2), \ldots$进行计数，这里$(i_t, c_t)$分别表示时刻$t$到达的项和它的计数增加值。对某个项$i \in [n]$，它在时刻$T$为止总的计数(频数)为：

$$f_i = Count(i, T) = \sum_{t: i_t = i, 1 \le t \le T} c_t$$

为此我们采用以下Count Sketch算法：首先从**全域**哈希函数族里随机选取哈希函数h: $[n] \rightarrow [k]$, 另外随机选取**全域**哈希函数g: $[n] \rightarrow \{-1,1\}$。

初始化：$C[1..k] \leftarrow 0$ (即k个计数器的值都初始化为0)

Process: (j, c) //当(j, c)到达时

$C[h(j)] \leftarrow C[h(j)] + c \cdot g(j)$

Output:

On query a, report $\widehat{f_a} = g(a)C[h(a)]$.

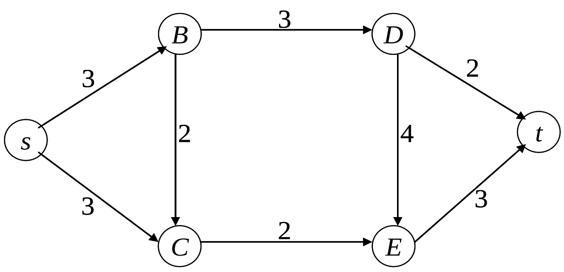
对任一项a，记$X = \widehat{f_a}$，计算X的期望值$E[X]$。(要求写出计算过程，只写出正确答案给2分)

3. 平摊分析 (10 分)

使用数组 $A[0..k\text{-}1]$来实现一个 k 位二进制计数器，初始化为全 0，在该数据结构上仅进行加一(Increment)操作，第 i 位翻转时的开销为 2^i (最低位为第 0 位，依此类推)，分析进行 n 次加一操作的平摊开销。(要求写出计算过程，只写出正确答案给 2 分)

4. 网络流 (12 分)

给定以下流网络，其中每条边上的数字为这条边的容量(capacity)。运行 Dinic 算法，依次写出每个阶段完成后找到的阻塞流(blocking flow)，以及最大流的值。



5. 动态规划 (18 分)

假如你收集了 n 首歌曲(歌曲编号为 1,2,…,n)，对于任意两首歌曲 a 和 b 你有个正的偏好分数 $pref[a,b]$，即播放了歌曲 a 后立刻播放歌曲 b 的偏好程度(这里 $pref[a,b]$不一定等于 $pref[b,a]$)，现在要求你使用动态规划来安排一个歌曲的播放顺序(每首歌恰好播放一次)，使得相邻的偏好分数**之和**最大。

子问题定义：给定这 n 首歌曲的任意一个子集 S 和 $i\in S$，记 $P[S,i]$为安排 S 中的歌曲以 i 为结尾，达到的最大相邻偏好分数之和。

a) 写出 $P[S,i]$的递归式，以及边界条件(即$|S| = 1$ 时)。 (7 分)

b) 以伪代码形式写出你的算法(要求返回最终的解)。 (8 分)

c) 分析你的算法的运行时间，并做简单解释。 (3 分)

6. NP 完全问题和线性规划 (25 分)

顶点覆盖(vertex cover)问题：给定一个无向图 $G = (V, E)$以及每个顶点上的非负权重 $w_v \geq 0$ ($v \in V$)，返回一个**最小权重顶点覆盖** $S \subseteq V$，即对任意一条边$(u, v) \in E$ 有 $u \in S$ 或 $v\in S$。

a) 用**整数**线性规划对此问题进行建模。(要求先定义决策变量，再写出目标函数和约束，6 分）

b) 给出将上述整数规划问题松弛为线性规划问题后的**对偶**问题。(6 分)

c) 写出顶点覆盖问题的判定版本，并解释该问题为 NP 问题。(3 分)

d) 选取一个你已知的 NPC 问题 (如 SAT, Clique, Independent set 等)，构造该问题到顶点覆盖问题的多项式时间归约。(需证明，10 分)

7. 近似算法 (15 分)

给定图$G = (V, E)$,若顶点子集$D \subseteq V$满足以下条件则称为**支配集(dominating set)**：对任意顶点 $u\in V$ 满足 $u\in D$ 或者存在$(u, v) \in E$ 的顶点 $v\in D$。最小支配集问题为 NP-难问题，以下是一个求支配集的近似算法：

$D \leftarrow \Phi$ (空集)

mark all vertices in V as “undominated”

while there are undominated vertices in V

  pick any undominated vertex v from V

  let N_v be the set containing v and all its neighbors

  $D \leftarrow D \cup N_v$

  mark all the vertices in N_v, and all the neighbors of vertices in N_v, as “dominated”

end while

a) 解释上述算法返回的集合 D 是一个支配集。 (2 分)

b) 证明任意支配集 AD 必须包含算法中所记的每个 N_v 中至少一个顶点。(5 分)

c) 记Δ为 G 中顶点的最大度数，即$\Delta = \max_{v\in V} \deg(v)$, 证明上述算法是近似度为$1+\Delta$的近似算法。(8 分)