高级算法设计与分析作业8
随机算法

傅彦璋 23S003008

2024.5.1

1

Algorithm 1: 随机折半查找

Data: 有序序列$a_1, ..., a_n$；待查找元素val

Result: val在序列中的位置，如val不存在于序列，则位置为-1

初始化$l \leftarrow 1$，$r \leftarrow n$;
while $l \leq r$ **do**
  均匀随机选取整数$k \in [l, r]$;
  if $a_k = val$ **then**
    return k
  end
  if $a_k > val$ **then**
    $r \leftarrow k - 1$;
  end
  if $a_k < val$ **then**
    $l \leftarrow k + 1$;
  end
end
return -1

随机折半查找算法的期望时间复杂度为$O(logn)$。

证明：

算法经历若干轮迭代，每一轮迭代将问题规模减小，使得问题规模从n逐步收缩至1。设随机变量T_n表示问题规模从n缩减到1经历的迭代次数。设随机变量X_n表示问题规模为n的时候，下一轮迭代使问题规模减小的程度。

存在某个单调非减函数$g(n)=E[X_n]$。我们用归纳法证明：$E[T_n]\le\int_1^n\frac{dx}{g(x)}$.记$f(n)=\int_1^n\frac{dx}{g(x)}$。假设对$m<n$，$E[T_m]\le f(m)$成立，往证$E[T_n]\le f(n)$。

$$
\begin{aligned}
E[T_n] &\le 1+E[T_{n-X_n}] \\
&\le 1+E\left[f(n-X_n)\right] \\
&= 1+E\left[\int_1^{n-X_n}\frac{dy}{g(y)}\right] \\
&= 1+E\left[\int_1^{n}\frac{dy}{g(y)}-\int_{n-X_n}^{n}\frac{dy}{g(y)}\right] \\
&\le 1+f(n)-E\left[\int_{n-X_n}^{n}\frac{dy}{g(y)}\right] \\
&\le 1+f(n)-E\left[\int_{n-X_n}^{n}\frac{dy}{g(n)}\right] \\
&\le 1+f(n)-\frac{E[X_n]}{g(n)} \\
&= f(n)
\end{aligned}
$$

所以，算法总共迭代轮数的期望$E[T_n]\le\int_1^n\frac{dx}{g(x)}$。现在，我们只需要找到$g(x)$。

接下来证明$g(x)\ge x/4$。考虑一轮迭代，当前问题规模为m（即算法第3行$[l,r]$区间的大小为m），下一轮问题规模为m_{next}。本次选择的k以$1/m$的概率等于$l,l+1,...,r-1$中的任何一个值，并将$[l,r]$划分为两个区间$[l,k-1]$和$[k+1,r]$。运气较坏时，下一轮的问题规模等于这两个区间中较长一个的长度。在这种“运气坏”的情况下，有：

$$
\begin{aligned}
E[m_{next}] &\le \frac{1}{m}\cdot 2\cdot\sum_{i=\lfloor m/2\rfloor}^{m-1} i \\
&\le \frac{1}{m}\left(\lfloor m/2\rfloor+m\right)\cdot\lceil m/2\rceil \\
&\le \frac{3m}{4}
\end{aligned}
$$

所以事实上，$E[m_{next}]\le\frac{3m}{4}$.所以$g(x)\ge x/4$。

于是，算法总共迭代轮数的期望$E[T_n]\le\int_1^n\frac{dx}{g(x)}\le 4\int_1^n\frac{dx}{x}=4\ln n\in O(logn)$。每轮迭代的操作只是随机选择一个位置$k$和比较大小，代价认为是$O(1)$的。所以随机折半查找算法的期望时间复杂度为$O(logn)$。

2

Algorithm 2: 计算$F(z)$

Data: z

Result: $F(z)$

均匀随机抽取$x \in \{0, 1, .., n-1\}$;

$y \leftarrow (z - x) \mod n$;

return $(F(x) + F(y)) \mod m$

Algorithm2能够以大于$\frac{1}{2}$的概率返回正确的$F(z)$。

这是因为，对$\forall x, P(F(x)\ is\ correct) = 0.8$。于是算法给出正确解的概率

$$P_{correct} = P(F(x)\ is\ correct\ and\ F(y)\ is\ correct) \geq 0.8^2 = 0.64 > \frac{1}{2}$$

如果算法运行3次：

如果算法3次给出的解相同，则返回该解。正确的概率$P_1 \geq 1-(1-P_{correct})^3 = 0.953344$.

如果算法2次给出的解相同，另一次不同，则返回2次相同的解。正确的概率$P_2 \geq 1-(1-P_{correct})^2 = 0.8704$.

如果运行3次结果均不同，则返回最后一次的解。正确的概率不低于0.64。

3

(1)

反证：假设I不是独立集，那么必然存在$u, v \in I$使得$uv \in E$。假设u的标签比v的小，那么随着算法的运行u一定比v更早加入集合I。当u加入集合I时，与其相邻的点，包括v在内，都会从S中删除。于是v永远不会被算法第4步取到，也永远不会加入I。$v \notin I$与假设矛盾。所以I一定是独立集。

(2)

结论有误。反例：考虑一个有三个点、两条边的连通图。题目中结论显然有误。

4

(1)

拉斯维加斯算法。

(2)

存在常数$b=0.75$满足题目要求。

证明（事实上证明过程与第1题证明$g(x)\geq \frac{x}{4}$的过程完全一致，只是符号表示不同）：

考虑某一轮递归，设此轮问题规模为m，随机变量m_{next}表示下一轮的问题规模。记本次选择的元素s是当前这轮m个数中第x小的，也就是说$x=|S_1|+1$。

如果$x=k$则下一轮问题规模为0；如果$x<k$则下一轮问题规模为$m-x$，如果$x>k$则下一轮问题规模为$x-1$。所以：

$$
\begin{aligned}
E[m_{next}] &= \frac{1}{m}\left(0+\sum_{x=1}^{k-1}(m-x)+\sum_{x=k+1}^{m}(x-1)\right)\\
&\leq \frac{1}{m}\left(\sum_{x=1}^{m/2}(m-x)+\sum_{x=m/2}^{m}(x-1)\right)\\
&= \frac{1}{m}\left(\left(m-1+\frac{m}{2}\right)\frac{m}{2}\right)\\
&= \frac{3m}{4}-\frac{1}{2}\\
&< \frac{3m}{4}
\end{aligned}
$$

所以，存在$b=\frac{3}{4}$使得算法递归过程中考虑的集合大小的数学期望不超过bn。

(3)

算法经历若干轮递归迭代，每一轮迭代将问题规模减小，使得问题规模从n逐步收缩至1。设随机变量T_n表示问题规模从n缩减到1经历的迭代次数。设随机变量X_n表示问题规模为n的时候，下一轮迭代使问题规模减小的程度。存在某个单调非减函数$g(n)=E[X_n]$。

我们在本次作业的第1题中已经证明，$E[T_n]\leq\int_1^n\frac{dx}{g(x)}$。(2)的结论告诉我们$g(n)\geq\frac{n}{4}$。于是$E[T_n]\leq 4\ln n$.第$i$轮迭代考虑的集合大小期望不超过$n\cdot\left(\frac{3}{4}\right)^i$.所以，算法运行时间期望

$$
E[Time]\leq\sum_{i=0}^{4\ln n}n\cdot\left(\frac{3}{4}\right)^i=4n\left(1-n^{-4\ln\frac{4}{3}}\right)\in O(n)
$$