دانشگاه صنعتی خواجه نصیرالدین طوسی

گزارش پروژه نهایی

درس طراحی الگوریتم

مسئله «شرکت پست»

مجتبی فیاضی – 9728973

توضیحات کلی

با داشتن N بسته با وزن‌های مختلف و K کامیون، هدف مسئله توزیع متناسب این بسته‌ها بین کامیون‌هاست بطوریکه وزن سنگین‌ترین کامیون حداقل شود. برای این منظور، از یک الگوریتم ساده حریصانه استفاده شده است. روال کار الگوریتم به این صورت است که بر روی بسته‌ها حلقه زده می‌شود و هر بسته به سبک‌ترین کامیون در آن لحظه اختصاص داده می‌شود.

```python
def applyGreedyDistribution(self):
    for item in self.packages:
        truck = self.trucks.get()
        truck.addPackage(item)
        self.trucks.put(truck)
```

شکل زیر یک نمونه اجرای الگوریتم بالا را با 10 بسته و 3 کامیون نشان می‌دهد.

```
Enter number of packages and trucks: 10 3
Enter package weights:
5 7 9 1 3 4 10 9 7 2
---------------------------
truck 2 => [(2, '7kg'), (6, '4kg'), (9, '7kg')] => 18kg
truck 3 => [(3, '9kg'), (7, '10kg')] => 19kg
truck 1 => [(1, '5kg'), (4, '1kg'), (5, '3kg'), (8, '9kg'), (10, '2kg')] => 20kg
Packages:
<1: 5kg>
<2: 7kg>
<3: 9kg>
<4: 1kg>
<5: 3kg>
<6: 4kg>
<7: 10kg>
<8: 9kg>
<9: 7kg>
<10: 2kg>
```

تحلیل الگوریتم

فرض کنید W وزن سنگین‌ترین کامیون در توزیع بالا باشد. می‌خواهیم نشان دهیم که مقدار W، از سنگین‌ترین وزن بهینه W^* خیلی بیشتر نیست. اما از آنجایی که مقدار W^* را نمی‌دانیم، باید حد پایینی برای آن پیدا کنیم. ایده‌های مختلفی برای حد پایین وجود دارد. یکی از این‌ها در نظر گرفتن وزن کل بسته‌ها است. یکی از K کامیون باید حداقل $1/K$ وزن کل را در خود جای دهد، بنابراین داریم:

$$W^* \geq \frac{1}{K}\sum_j w_j$$

که در آن w_j وزن بسته‌هاست. علاوه بر این می‌دانیم که وزن بهینه W^* بزرگتر یا مساوی سنگین‌ترین بسته است:

$$W^* \geq max w_j$$

با داشتن دو مورد بالا، می‌خواهیم ثابت کنیم که الگوریتم ارائه شده یک توزیعی با $W \leq 2W^*$ بدست می‌دهد.

برای اثبات این رابطه به این صورت عمل می‌کنیم. طبق الگوریتم می‌دانیم وقتی بسته j ام (بسته آخر) را به کامیون i ام اختصاص دادیم، این کامیون سبک‌تر از همه بود (نکته اساسی الگوریتم حریصانه بالا). وزن این کامیون قبل از اختصاص این بسته برابر $W_i - w_j$ بود و از آنجایی که این کامیون از همه کامیون‌ها سبک‌تر بود، می‌توان نتیجه گرفت که وزن همه کامیون‌ها در این زمان حداقل $W_i - w_j$ است.

بنابراین برای جمع وزن همه کامیون‌ها رابطه زیر برقرار است:

$$W_i - w_j \leq \frac{1}{K} \sum_k W_k \quad \text{یا} \quad \sum_k W_k \geq K(W_i - w_j)$$

اما $\sum_k W_k$ همان جمع وزن تمام بسته‌هاست ($\sum_j w_j$)، بنابراین سمت راست نامعادله سمت چپ دقیقاً حدپایین جواب بهینه است. پس داریم:

$$W_i - w_j \leq W^*$$

حالا به بسته نهایی را وارد محاسبات میکنیم. اگر وزن بسته نهایی w_j را به کامیون آخر اضافه کنیم، با توجه به رابطه $W^* \geq \max w_j$ می‌توان نتیجه گرفت:

$$W_i = (W_i - w_j) + w_j \leq 2W^*$$

از آنجایی که W_j همان W است، بنابراین ثابت کردیم که نرخ تقریب الگوریتم ما برابر 2 است:

$$W \leq 2W^*$$

پیچیدگی زمانی

الگوریتم یک لوپ اصلی به اندازه N دارد که در هر تکرار آن، یک عملیات افزودن و یک عملیات حذف از صف اولویت کامیون‌ها انجام می‌شود، بنابراین پیچیدگی زمانی برنامه برابر است با:

O(N logK)