



جست‌وجوی سطح‌اول

تعریف

جست‌وجوی سطح‌اول سطح (سطح‌اول) که به s مشهور است، روشی برای پیمایش گراف است که بعد از جست‌وجوی عمق‌اول مهم‌ترین و پرکاربردترین الگوریتم پیمایش گراف است. این الگوریتم معمولاً در گراف‌های وزن دار کاربرد خاصی ندارد و عمده نقش آن در گراف‌های بی‌وزن و برای پیدا کردن کوتاه‌ترین فاصله بین یک رأس تا بقیه‌ی رأس‌هاست.

الگوریتم

این الگوریتم ابتدا یک رأس ریشه مانند s را به عنوان شروع می‌گیرد. سپس رأس‌ها را سطح بندی میکند. سطح بندی به این صورت است که تمام رأس‌های مجاور s را در سطح اول قرار می‌دهیم، حال برای سطح $1 + i$ ، رأس i که مجاور یکی از رئوس سطح i است را در این سطح قرار می‌دهیم به شرطی که در هیچ سطح دیگری نیامده باشد. به عبارت دیگر رأس‌ها را بر اساس کوتاه‌ترین فاصله از s سطح بندی میکنیم. حال برای پیمایش به ترتیب سطح، و در هر سطح به ترتیب دلخواه وارد رأس‌ها میشویم. پس اولین زمانی که به هر رأس می‌رسیم، با کمترین فاصله ممکن نسبت به s رسیده ایم.

پیچیدگی الگوریتم

با توجه به اینکه به هر رأس حداکثر یکبار وارد میشویم در نتیجه هر یالی هم حداکثر دوبار به ازای دو سر آن شمرده می‌شود، پس در کل هزینه الگوریتم از $O(n + e)$ است که n تعداد یال‌ها و e تعداد رأس‌ها است.

پیاده‌سازی

برای ایجاد سطح‌های مختلف و حرکت به ترتیب سطح‌ها نیاز به استفاده از صف داریم. فرض کرده ایم که لیست مجاورت رأس‌ها را داده اند و الگوریتم رأس‌ها را از آغازین تا آخرین از ورودی می‌گیرد.

```
#include <queue>
#include <vector>

const int MAXN = 100 * 1000 + 10;

bool mark[MAXN];
int vector<int> adj[MAXN];

void bfs(int v) {
    queue<int> q;

    mark[v] = 1;
    q.push(v);

    while(q.size()) {
        v = q.front(); // رأس ابتدایی را از صف بر می‌داریم
        q.pop();
        // کارهای پیش‌ترتیب را اینجا مینویسیم
        for(int i = 0; i < adj[v].size(); i++) {
            int u = adj[v][i];

            if (mark[u] == 1) // قبلاً این رأس را به صف اضافه نکرده باشیم
                continue;

            mark[u] = 1;
            q.push(u);
        }
    }
}
```

}

- تابع اوایلر - سایت ماکزیمال [الومداری مچن قریل اورد امد - ح- قریل ایز معبرا]
- تابع اوایلر - ویکی پدیا [الومداری مچن قریل اورد امد - ح- قریل ایز معبرا]