

第 3-X 讲-订正

姓名: 林凡琪 学号: 211240042

评分: _____ 评阅: _____

2023 年 2 月 5 日

请独立完成作业，不得抄袭。
若得到他人帮助，请致谢。
若参考了其它资料，请给出引用。
鼓励讨论，但需独立书写解题过程。

1 作业 (必做部分)

2 订正

题目 1 (3-12.3 CZ 8.14)

Prove that a graph G without isolated vertices has a perfect matching if and only if $\alpha_1(G) = \beta_1(G)$.

解答:

如果 n 阶图 G 有完美匹配, 那么最大匹配数 $= \beta_1 G = \frac{n}{2}$.

又因为 $\beta_1 G + \alpha_1 G = n$,

所以可得 $\alpha_1(G) = \beta_1(G)$.

如果 $\alpha_1(G) = \beta_1(G)$, 那么 $\alpha_1(G) = \beta_1(G) = \frac{n}{2}$

所以图的最大匹配数为 $\frac{n}{2}$ 且完美匹配是存在的.

综上得证.

题目 2 (3-12.4 CZ 8.16)

Prove that if G is a graph of order n , maximum degree δ and having no isolated vertices, then $\beta(G) \geq \frac{n}{\delta+1}$

解答:

设 G 的最小点覆盖集为: S , 同时 T 为 $G - S$.

由点覆盖集性质知, T 包含了 S 在 G 中的全部所连点, 所以可知 $|T| \leq \delta \times |S|$, 那么

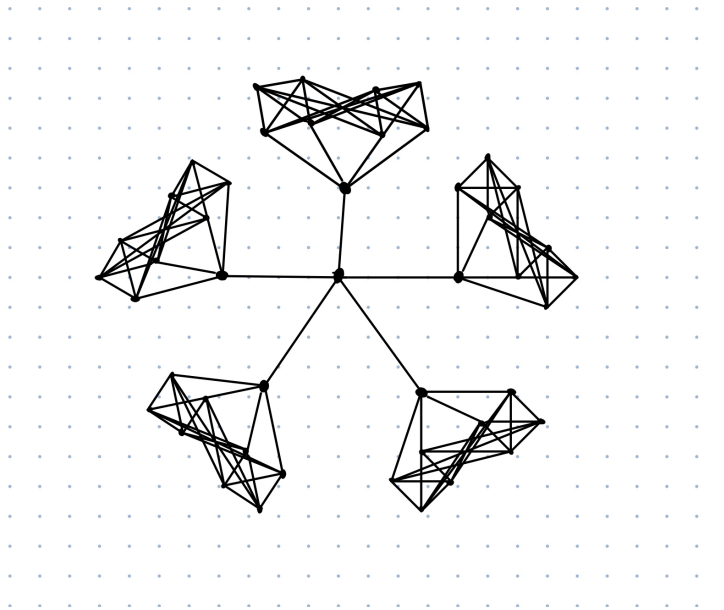
$$n - \beta(G) \leq \delta \times \beta(G)$$

最后可得 $\beta(G) \geq \frac{n}{\delta+1}$

题目 3 (3-12.5 CZ 8.18)

Give an example of a 5-regular graph that contains no 1-factor.

解答:



题目 4 (3-7.1 TC 22.1-3)

The **transpose** of a directed graph $G = (V, E)$ is the graph $G^T = (V, E^T)$, where $E^T = \{(v, u) \in V \times V : (u, v) \in E\}$. Thus, G^T is G with all its edge reversed. Describe efficient algorithms for computing G^T from G , for both the adjacency-list and adjacency-matrix representations of G . Analyze the running times of your algorithms.

解答:

For adjacency tables, each point, each edge is traversed constantly, and the time

```

1: procedure ADJACENCY-LIST ( $V$ )
2:   for  $x \in V$  do
3:     for  $y \in E(x)$  do
4:       add  $x$  to  $E^1(y)$ 
5:     end for
6:   end for
7: end procedure

```

Algorithm 1: adjacency-list

```

1: procedure ADJACENCY-MATRIX ( $V$ )
2:   for  $x \in V$  do
3:     for  $y \in V$  do
4:       if  $(x, y) \in E$  then
5:         add  $(y, x)$  to  $E^1$ 
6:       end if
7:     end for
8:   end for
9: end procedure

```

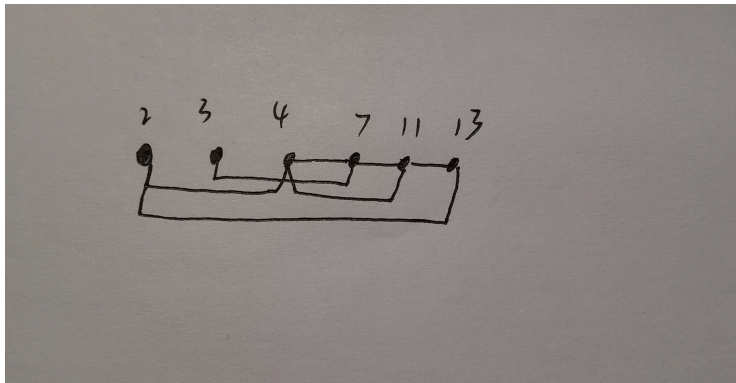
Algorithm 2: adjacency-matrix

complexity $O(V + E)$ For adjacency matrices, double loop, $O(V)$ per layer, $O(V^2)$ in time complexity

题目 5 (3-4.2 CZ 1.3)

Let $S = \{2, 3, 4, 7, 11, 13\}$. Draw the graph G whose vertex set is S and such that $ij \in E(G)$ for $i, j \in S$ if $i + j \in S$ or $|i - j| \in S$.

解答:

**题目 6 (3-4.6 CZ 2.1)**

Give an example of the following or explain why no such example exists:

- (a) a graph of order 7 whose vertices have degrees 1, 1, 1, 2, 2, 3, 3.
- (b) a graph of order 7 whose vertices have degrees 1, 2, 2, 2, 3, 3, 7.
- (c) a graph of order 4 whose vertices have degrees 1, 3, 3, 3.

解答:

- (a) Non-existent. Its sum of degrees is 13, which is not even and contrary to the first law of graph theory.
- (b) Non-existent. For a figure with a point of 7, the maximum degree is 6.
- (c) Non-existent. A figure with a point of 4 has a degree of 3, indicating that if 3 points are connected to all other points, it is impossible to have a point with a degree of 1.

题目 7 (3-2.1 TC 16.1-2)

Suppose that instead of always selecting the first activity to finish, we instead select the last activity to start that is compatible with all previously selected activities. Describe how this approach is a greedy algorithm, and prove that it yields an optimal solution.

解答:

Assuming that all event intervals are left open and right closed, the algorithm proposed in the textbook is obviously still correct.

Suppose the original i activity starts at s_i and ends at t_i . Let $T = \max t_i$.

Reset the timeline in reverse order. That is, under the new timeline, the start time of each activity $s'_i = T - t_i$, and the end time $t'_i = T - s_i$.

The event time under the original problem is $[s_i, t_i)$, and the event time under the new problem is $(t'_i, s'_i]$.

Obviously, for this problem, after conversion, the problem is still the same as the original problem and has the same optimal solution.

After conversion, according to the proven algorithm, the event with the earliest end is prioritized each time. At this point, the event that ended first is the same as the event that first started the original problem. Selecting the event that ends earliest after the transition is equivalent to selecting the event that starts later in the original issue.

Therefore, this approach can still obtain the optimal solution.

3 反馈