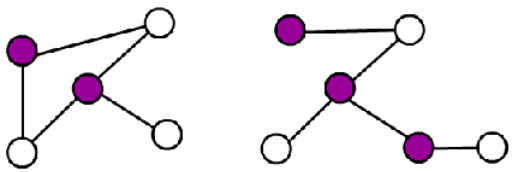
Title Title Title Title Title

author
xxxxxx@xxx.com
2022 年 3 月 19 日

Block & Image[1]

Definition

A vertex cover of a graph is a set of vertices such that each edge of the graph is incident to at least one vertex of the set.



Minimum Vertex Cover

[1] footnote

Code Example

```
bool dfs(int x) {
    for (auto &y:e[x])
    if (!used[y]) {
        used[y] = true;
        if (link[y] == -1 || dfs(link[y])) {
            link[y] = x;
            return true;
        }
    }
    return false;
}
```

```
int hungary() {
    int res = 0;
    memset(link, -1, sizeof(link));
    for (int i = 1; i <= n; ++i) {
        memset(used, false, sizeof(used));
        if (dfs(i)) ++res;
    }
    return res;
}
```

Pseudocode

- numerical analysis, computational geometry...
- time complexity $O(\log((R-L)/\epsilon))$

```
1: procedure BinarySearchOnRealNumbers(L, R)
2:     while R − L > ϵ do
3:         mid ← (L + R)/2
4:         if lessThanAns(mid) then
5:             L ← mid
6:         else
7:             R ← mid
       return L
```

Puase

- 测试一下pause语句的功能

任务

统计及格学生的平均成绩 (不计不及格的学生和分数)

```
scores = [76, 83, 89, 45, 67, 89, 85,
    77]
sum_, count = 0, 0
for score in scores:
    if score < 60:
        continue
    sum_ += score
    count += 1
print('平均成绩为:', sum_/count)
```

```
scores = [76, 83, 89, 45, 67, 89, 85,
    77]
sum_, count = 0, 0
for score in scores:
    if score >= 60:
        sum_ += score
        count += 1
print('平均成绩为:', sum_/count)
```

Puase

- 测试一下pause语句的功能

任务

统计及格学生的平均成绩 (不计不及格的学生和分数)

```
scores = [76, 83, 89, 45, 67, 89, 85,
    77]
sum_, count = 0, 0
for score in scores:
    if score < 60:
        continue
    sum_ += score
    count += 1
print('平均成绩为:', sum_/count)
```

```
scores = [76, 83, 89, 45, 67, 89, 85,
    77]
sum_, count = 0, 0
for score in scores:
    if score >= 60:
        sum_ += score
        count += 1
print('平均成绩为:', sum_/count)
```

Puase

- 测试一下pause语句的功能

任务

统计及格学生的平均成绩 (不计不及格的学生和分数)

```
scores = [76, 83, 89, 45, 67, 89, 85,
    77]
sum_, count = 0, 0
for score in scores:
    if score < 60:
        continue
    sum_ += score
    count += 1
print('平均成绩为:', sum_/count)
```

```
scores = [76, 83, 89, 45, 67, 89, 85,
    77]
sum_, count = 0, 0
for score in scores:
    if score >= 60:
        sum_ += score
        count += 1
print('平均成绩为:', sum_/count)
```

Puase

- 测试一下pause语句的功能

任务

统计及格学生的平均成绩 (不计不及格的学生和分数)

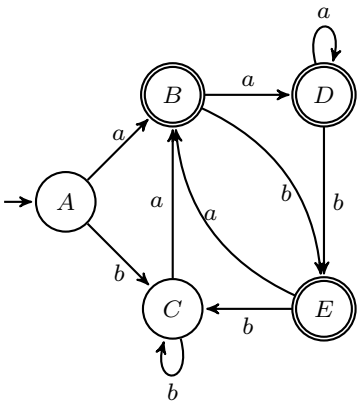
```
scores = [76, 83, 89, 45, 67, 89, 85,
    77]
sum_, count = 0, 0
for score in scores:
    if score < 60:
        continue
    sum_ += score
    count += 1
print('平均成绩为:', sum_/count)
```

```
scores = [76, 83, 89, 45, 67, 89, 85,
    77]
sum_, count = 0, 0
for score in scores:
    if score >= 60:
        sum_ += score
        count += 1
print('平均成绩为:', sum_/count)
```

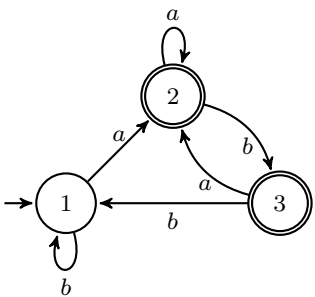
任务

将正则表达式 $(a \mid b) * a(a \mid b \mid \epsilon)$ 转化成一个 DFA 并最小化

先构造出 NFA，之后使用**子集构造法**转 DFA



最小化 $1 : \{A, C\}, 2 : \{B, D\}, 3 : \{E\}$



Math Formula

本页的字体大小为 12pt，vskip 为 14pt

- 欧拉公式：

$$e^{i\pi} + 1 = 0$$

- 拉格朗日插值：

$$A(x) = \sum_{k=0}^{n-1} y_k \frac{\prod_{j \neq k}(x - x_j)}{\prod_{j \neq k}(x_k - x_j)}$$

Table

Algorithm	Time (worst)	Time (avg.)	Space	Stable	In-place
insertion sort	$\Theta(n^2)$	$\Theta(n^2)$	$O(1)$	yes	yes
merge sort	$\Theta(n \log n)$	$\Theta(n \log n)$	$O(N)$	yes	no
heapsort	$O(n \log n)$	$O(n \log n)$	$O(1)$	no	yes
quick sort	$\Theta(n^2)$	$\Theta(n \log n)$	$O(N)$ $O(\log N)$	no	yes
counting sort	$\Theta(k+n)$	$\Theta(k+n)$	$O(k)$	yes	no
radix sort	$\Theta(d(k+n))$	$\Theta(d(k+n))$	$O(k+n)$	-	no
bucket sort	$\Theta(n^2)$	$\Theta(n)$	$O(n)$	yes	no

Other Tables

i	0	1	2	3	4	5
s	$	#	a	#	a	#

表 1: A realistic table I have built

Small col	Big col		
Grouped items	Item 1		
	Item 2		
Usual row	Spam	Bacon	Eggs