1. 搜索
2. Flood Fill（泛洪算法）

填充算法，可DFS也可BFS（大图必须BFS，不然栈要爆炸）。

1. 最短路模型

可DFS也可BFS（大图必须BFS，不然栈要爆炸）。

1. 多源BFS

将所有源点加入队列，然后直接开搜。

1. 最小步数模型

求某状态到某状态的最小操作步数。只需知道怎么将状态哈希化，然后BFS开搜即可。

1. 双端队列BFS

图边权为0或1，计算两点最短路径。可用双端队列，与0入队首，遇1入队尾，这样每次队首出来时必得到该点最短路径。

1. 双向BFS

搜索范围过大，但起始状态和终止状态已知，那么可以从起始状态和终止状态同时开始搜索。

1. A\*算法

在最短路问题中，当所有边权值皆非负时，可用启发函数来优化BFS过程。当点数很多，多到不能把这些点之间的最短距离算出来时适用。BFS的队列用优先队列，排序依据是起点到当前点所花费+当前点到终点预计花费。

1. DFS剪枝与优化

优化搜索顺序：确定搜索顺序，使该顺序能覆盖所有方案，且优先搜索分支较少的节点；排除等效冗余；可行性剪枝：根据题目给出的限定条件进行剪枝；最优性剪枝：将比目前已知方案更差的方案剪掉；记忆化。

1. 迭代加深DFS

若答案与搜索深度有关，但不知道搜索深度，可从小到大枚举搜索深度，对每个搜索深度去跑DFS。

1. 双向DFS

搜索中，可以把要搜索的集合一分为二，一半在DFS中打成表；另一半也DFS，在叶子节点快速（如二分法）从表中找到与当前节点匹配的最优解作为当前最优方案。该方法可将算法时间复杂度从O（c^n）优化到O（c^n/2 \* m）（m为查表的时间复杂度）。

1. IDA\*算法

A\*算法与迭代加深DFS的结合。在迭代加深DFS中，当估计值和当前值相加大于max\_depth时剪枝，当估计值为0时到达终点。