

BUAA 数学建模 L^AT_EX 模板

摘要

cumcmthesis 是为全国大学生数学建模竞赛编写的 L^AT_EX 模板, 旨在让大家专注于论文的内容写作, 而不用花费过多精力在格式的定制和调整上. 本手册是相应的参考, 其中提供了一些环境和命令可以让模板的使用更为方便. 同时需要注意, 使用者需要有一定的 L^AT_EX 的使用经验, 至少要会使用常用宏包的一些功能

关键字: 状态空间法 商人过河 广度优先搜索 哈希去重优化

目录

一、问题重述	3
1.1 解决方案	3
二、模型的假设	3
2.1 商人过河	3
三、符号说明	4
四、插入图片	4
4.1 问题二分析	4
五、绘制普通三线表格	4
参考文献	6

一、问题重述

商人过河问题：三名商人各带一名仆人渡河，一只小船只能容纳二人，由他们自己划行。仆人们密约，在河的任一岸，一旦仆人数比商人多，就杀人越货。但如何乘船渡河的大权掌握在商人们手中，商人怎样才能安全渡河呢？

1.1 解决方案

商人过河问题：商人过河问题可以视为一个多步决策过程：船由此岸驶向彼岸或从彼岸驶回此岸，都要对船上的人员数量（商人、仆人数）做出决策，在保证安全的前提下（即两岸的商人数均不少于仆人数），经过一系列有限步骤使人员全部过河。我们首先建立模型，将商人和仆人的可行组合抽象成二元组 S ，将每次过河的商人和仆人组合也抽象成一个二元组 D ，于是问题归结为：寻找一种搜索策略，使二元组 S 从初始状态经过有限且安全步骤后达到目标状态。我们运用图解法快捷直观地对于问题进行解决与推广，并用 C++ 语言代码对问题进行验证，完善了对于商人过河问题的求解。

二、模型的假设

2.1 商人过河

- 假如在河岸的某一侧只有随从没有商人，随从并不会将货物拿走或者逃走。
- 假设小船停到了船的一岸，并不会在没有人划船的情况下来到另一岸。
- 无论船上有几个商人和随从，随从都不会杀人越货。

三、符号说明

符号	意义
x_i	第 i 次船到岸时候，出发岸上的商人数量
y_i	第 i 次船到岸时候，出发岸上仆人的数量
$s_i = (x_i, y_i)$	第 i 次船离开时候，出发岸上的状态
u_i	第 i 次船到岸时候，船上的商人数量
v_i	第 i 次船到岸时候，船上的仆人数量
$d_i = (u_i, v_i)$	第 i 次船到岸时候，船上的状态
n	拓展研究中商人和仆人的总人数
k	拓展研究中船的容量

四、插入图片

4.1 问题二分析

八数码问题

$$wo = gongshi \quad (1)$$

问题流程图：

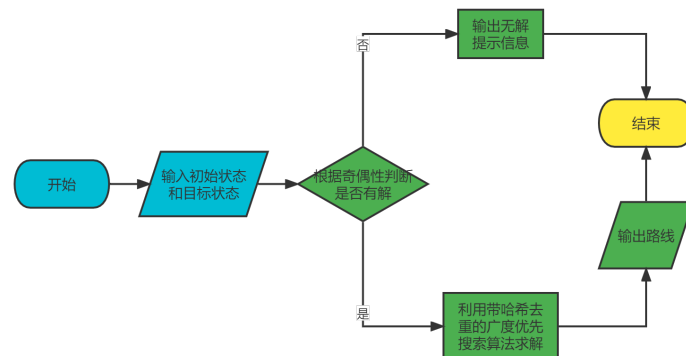


图 1 问题二流程图

五、绘制普通三线表格

表格应具有三线表格式，因此常用 booktabs 宏包，其标准格式如表 1 所示。
插入代码案例：其绘制表格的代码及其说明如下。

表 1 标准三线表格

$D(\text{in})$	$P_u(\text{lbs})$	$u_u(\text{in})$	β	$G_f(\text{psi.in})$
5	269.8	0.000674	1.79	0.04089
10	421.0	0.001035	3.59	0.04089
20	640.2	0.001565	7.18	0.04089

```

\begin{table}[!htbp]
\caption[标签名]{中文标题}
\begin{tabular}{cc...c}
\toprule[1.5pt]
表头第1个格 & 表头第2个格 & ... & 表头第n个格 \\
\midrule[1pt]
表中数据(1,1) & 表中数据(1,2) & ... & 表中数据(1,n)\\
表中数据(2,1) & 表中数据(2,2) & ... & 表中数据(2,n)\\
.....\\
表中数据(m,1) & 表中数据(m,2) & ... & 表中数据(m,n)\\
\bottomrule[1.5pt]
\end{tabular}
\end{table}

```

`table` 环境是一个将表格嵌入文本的浮动环境。`tabular` 环境的必选参数由每列对应一个格式字符所组成：`c` 表示居中，`l` 表示左对齐，`r` 表示右对齐，其总个数应与表的列数相同。此外，`@{文本}` 可以出现在任意两个上述的列格式之间，其中的文本将被插入每一行的同一位置。表格的各行以 `\\` 分隔，同一行的各列则以 `&` 分隔。`\toprule`、`\midrule` 和 `\bottomrule` 三个命令是由 `booktabs` 宏包提供的，其中 `\toprule` 和 `\bottomrule` 分别用来绘制表格的第一条（表格最顶部）和第三条（表格最底部）水平线，`\midrule` 用来绘制第二条（表头之下）水平线，且第一条和第三条水平线的线宽为 1.5pt，第二条水平线的线宽为 1pt。引用方法：“如表 `\ref{标签名}` 所示”。

参考文献

- [1] 张念发, 张宪新, and 刘长征, “基于状态空间搜索法的商人过河问题解决方案,” 电脑编程技巧与维护, no. 18, pp. 36–37, 2010.
- [2] 詹志辉, 胡晓敏, and 张军, “通过八数码问题比较搜索算法的性能,” 计算机工程与设计, vol. 28, no. 11, pp. 2505–2508, 2007.
- [3] 钱莹, “基于广度优先搜索的八数码问题解决方案,” 电脑学习, no. 1, pp. 45–46, 2008.
- [4] 付宏杰, 王雪莹, 周健, 周孙静, 朱珠, and 张俊余, “八数码问题解法效率比较及改进研究,” 软件导刊, vol. 15, no. 9, pp. 41–45, 2016.
- [5] 周浩, “八数码问题 dfs 和 bfs 算法的设计与实现,” 电脑知识与技术: 学术交流, vol. 7, no. 8, pp. 5487–5489, 2011.