

基于在线裁判的编程学习系统：设计思路

孙秋实

时间管理超级大师组

版本：1.00

日期：2020 年 8 月 21 日

摘 要

本文简单解释基于在线裁判的编程学习系统中教程式学习和推荐式学习的实现思路，以及一些在组会中讨论过的零散问题。

关键词：one-hot 编码，词袋模型

1 界面设计

1.1 界面划分

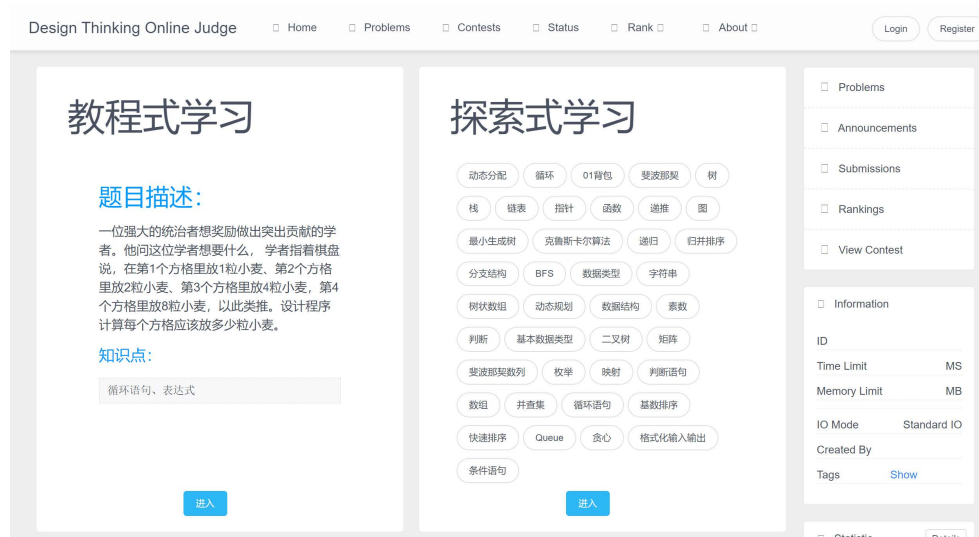


图 1: 暂定界面结构

暂定两个入口，学生可以自由选择（如果没有任何答题记录的话可以选择教程式学习冷启动），除此之外原有的 OnlineJudge 功能不变

2 系统结构 (在原系统上改进)

题库	ProblemID+Difficulty+label+ProbStatus(Done?)		
用户	UserID+Accuracy+UserStatus		
	UserStatus	取决于推荐系统需求	

图 2: 需要在原 OJ 系统上进行增补的元数据

3 教程式学习

简单称之为假推荐

3.1 设置教程式学习的目的

- 1). 服务无经验新手
- 2). 解决冷启动问题

3.2 需要采集的数据

- 1). 用户在一定量习题中的准确率 → 辅助难度标签
- 2). 用户频繁提交出错的习题类型 → 辅助知识点标签
- 3). 将教程式推荐已经涉及过的题目从推荐列表中剔除

3.3 题目跳转逻辑 (举例)

输入直角坐标 (以字符 q 结束), 将其转化为极坐标并输出。

标签: 函数

极坐标用向量的模 (即向量的长度) 和向量相对 x 轴逆时针旋转的角度来描述该向量。直角坐标用向量的 x 轴和 y 轴的坐标来描述该向量。编写一个程序, 读取向量的模和角度 (单位: 度), 然后显示 x 轴和 y 轴的坐标。相关方程如下: $x = r \cdot \cos A$; $y = r \cdot \sin A$ 。编写一个函数来完成转换。该函数接受一个包含极坐标的结构, 并返回一个包含直角坐标的结构。输入极坐标, 输出直角坐标。

标签: 函数, 结构体

流程: 用户完成一道题后跳转至另一道标签有重合的习题, 以此类推, 直到有该标签的习题被全部完成, 再启动新的知识点 (可以先准备下伪代码)

4 真推荐式学习

4.1 基于内容的推荐系统所需数据形式

1). 独热编码 (one-hot)

1). 难度标签 e.g. $v_{difficulty}=[0,0,1]$ ([0,0,1] 表示难, [easy,normal,hard])

2). 题目状态标签 e.g. $v_{status}=[1,0,0]$ ([1,0,0] 表示难, [New,Done,WrongAns])

2). BOW 模型(简化举例): $v_{BOW}=[\text{变量类型}, \text{函数}, \text{指针}, \text{结构体}, \text{文件}]$ ∴ e.g. $v_{BOW}=[0,1,1,1,0]$ 表示该题同时考察函数, 指针, 结构体三个知识点

4.2 数据在数据库中的存储形式

4.2.1 方案 1

新增 ProblemID 和一个装载所有推荐系统数据形式的信息 e.g:

ProblemID	Characteristics
996	1,0,0,1,0,1,0,1,1,0,0,0,1
251	1,0,0,1,0,1,0,0,0,0,1,0,1,0

4.2.2 方案 2

新增 ProblemID 和两个独热编码和一个词袋模型关系, 可以单独调用/加权 e.g:

ProblemID	difficulty	label	status
996	1,0,0	1,0,1,0,1,1,0	0,0,1
251	1,0,0	1,0,1,0,0,0,1	0,1,0

Remarks: 最终选择哪种方案基于推荐算法的设计

5 问题

1. OJ 数据库增删改查是否能完全摸清楚
2. 网页之间依赖数据库中标签项的跳转是否能完全解决
3. 基于内容推荐算法还没有进行过测试