**OJMOOC背景**

1. **概述**

OJMOOC是一个为中学生提供的集MOOC教学，OnlineJudge，在线辅导于一身的在线编程学习平台，致力于优化中学生编程学习方式，提高学习的效果。

1. **为什么需要OJMOOC？**

世界已经跨过了21世纪的第一个十年，互联网与计算机技术以一种前所未有的速度渗透到了每一个人的生活之中。在世纪初计算机刚开始普及的时候，一个人完全可以脱离互联网正常生活。但是仅仅过了十年，计算机以及各种手持设备开始毫无死角地覆盖到人们的生活中，互联网成为了每个人的生活中不可分离的一部分。同时随着各种设备的普及，人们对于软件的使用也较十年以前有了很大的变化。十年前，大部分人仅仅是用浏览器看一些静态网页，用编辑器写文本，最多再玩一些游戏。一切都停留在“有电脑真方便”这样的概念上，将其作为某些场景下提高生产力的工具。但如今，软件已然成为了绝大多数行业提高生产力的支柱工具，离开了软件不再是降低效率的问题，甚至整个体系都将无法运行。软件已经成为了这个世界的运行之本。

各行各业对于软件产生了重度的依赖之后，随之而来的问题至少有两个：

* 1. 对于软件的功能性有着前所未有的高要求。

当软件成为了根本，就意味着原先很多由人工负责的工作都将由软件来完成，这就产生了需求的扩展。而软件带来的生产力解放又会让更多人去开拓新的领域和市场，从而会孕育出全新的软件需求。

* 1. 之后大部分的生产力提高都将基于软件来改进。

人类社会的发展归根结底是取决于生产力的发展水平，而人们总会不知疲倦地去寻找提高生产力的方法。随着摩尔定律的时间间隔越来越长，一直蓬勃发展的计算机硬件行业可能已经开始进入了一个瓶颈期。而与此同时，全世界软件的发展却进入了一个快速上升期，下一个时代的生产力提高点之一毫无疑问就是软件。对于企业来说就是开发新的软件平台，从而进一步解放人工，提高运行效率；对于个人来说，就是学习使用各种软件来提高个人的工作效率，减轻多余的负担。生产力的提高很大程度上将基于软件。

由此可以看来，今后人们的生活将和软件紧密地绑定在一起，软件也不再仅仅是一个工具的概念，而是关系到生产力的重要支柱。因此如何教人们编写软件，而不仅仅是使用，成为相当重要的议题。而这种教育的最佳开始，就是中学生。这就是我们OJMOOC项目的初衷。

**2、为什么中学生要学习编程？**

中学生是最适合学习编程的群体之一。对于小学生而言，最好还是以学习软件的使用为主，编程的很多概念比较复杂不容易理解。对于已经工作的成年人而言，一方面已经过了学习的最佳时期，另一方面可能并没有太多的时间来研究编程的技术和方法。

1. **传统的课堂教育不行吗？**

传统的课堂教育在编程教育上还不够好。我本人在中学的时候就上过这样的课程。在教授理论的时候，老师只是在课堂上说，黑板上写，课后布置几道题目就结束了。除了知道几个技术名词之外对于程序本身并没有多少理解，判断循环什么的概念只是知道，但也不知道要怎么用，或者干什么用的。到了实践课程的时候，老师就在机房黑板上先写好这节课要干什么，说个大概的步骤或思路，然后同学们开始照着黑板上的步骤开始操作。其中如果有问题就只能互相打听或者等老师巡逻过来再提问。这里隐藏的一个问题是，如果一个学生一开始就没有听懂，或是干脆什么都不会，他（她）会完全跟不上。就是去问老师也不知道该问什么，因为根本就无从下手。

这里就体现出在技术教育上传统课堂的弊端。理论知识好歹还能够流于一下形式，实践知识就很难有保障，一个老师这么多学生，怎么教？课后作业的批改也是一个问题，传统的作业往往老师最多机械性的看一下对错或者是写的好不好就可以了。但是对于编程的课程，一个作业对不对往往还涉及到这个程序编译过不过，运行是不是正确的问题，不仅仅是看一下注释逻辑对不对就可以的了。因此在编程教育的领域上需要新的教学方式。

1. **MOOC和OJ不是已经有人在做了吗？**

新的教学方式其实很早就出现了。其中一种就是视频教育。比如这两年很火的公开课，各个大学都推出了自己的公开课内容，其中不乏许多优质的程序设计课程。这些课程的主要缺点是缺乏一个比较好的判定体系，很容易让学习者养成听过算过的习惯。真正上完课程，做完全部作业的人占所有听课者的比例其实非常的低。作为公开课的改进，MOOC出现了。在视频教育的基础上添加了评判系统等等。比如Coursa就提供了peer assessments evaluation，让学生可以互相判卷，达到检验学习效果的目的。但是对于编程类的课程而言，光是互相判卷还不够。很多时候需要将所编写的程序编译运行来验证正确性，互相编译验证不是不可以，只是效率和可靠性非常的低。真正需要的是一个自动的编译验证系统，即Online Judge.

说到Online Judge也不是什么新鲜的东西了。自从ACM竞赛出现以来，各种OJ系统就层出不穷。主要流程基本就是选择语言，输入源码，服务器编译执行，返回结果。这样的一个模式基本就能做出一个OJ来了。但是这些OJ系统最大的一个问题是：默认用户都是经验者，即知道一个程序应该怎么写才是对的。不知道怎么写程序的人想来也不会用OJ。但这对于一个中学生来说就比较困难了，因为编程经验是一张白纸的人大有人在。如果自己写的程序OJ验证不通过会极大增加变成学习过程中的挫败感，这对于提高他们的学习效果实际上起到的是反作用。学生们需要的往往是一个手把手的教学，而不是被直接扔到一个技术的森林里任其自生自灭。

1. **OJMOOC是如何解决这些问题的？**

前面说到，对于编程的课程来说是缺少一个验证的系统。而对于OJ来说，是缺少一个合适的引导。因此解决的方法其实非常的简单：把两者结合起来就可以了。OJ+MOOC = OJMOOC，在线课程的基础上提供程序执行功能的一个项目，能让学生跟着老师学习，改写或执行老师写好的代码来观察结果，获得对于程序执行最直观的感受。最简单的情况下，只需要点击一下执行就能运行老师已经写好的代码，这就相当于把编程的门槛直接拉低了一个级别，从而让学生们可以更好的上手。这就是OJMOOC解决编程学习问题的基本思路。

1. **OJMOOC现在有需求吗？**

有。全国信息学奥林匹克竞赛（NOI）就是一例。学校在备战NOI的时候往往会遇到的问题一方面是教师数量不足，师生比较其他学科要小得多，老师很难做到很细致的辅导。另一方面就是配套的设施并不完备，比如最多只能做出课堂录制的视频，让学生们回放观看。对于程序的正确与否也只能由学生自己去编译执行，没有一个统一的系统去做这样的事，这对掌握学生的学习进度实际上是一个挑战。

OJMOOC就为老师提供了课程录制的功能，甚至还包括了程序的代码，这样学生就可以直接使用老师的代码运行查看结果而不用切换到自己的IDE中编写代码，节省了上下文切换的时间。