

程序员最核心的能力是什么？

作者: [Hefery](#)

原文链接: <https://ld246.com/article/1642529858220>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

需求转换的能力

需求评审（工作量、优先级） --> 系统设计、技术评审 --> 开发、测试（各环境测试报告） --> 提（代码相关资源、资源清单、资源对比）

理解需求，沟通+逻辑

- 没有沟通：接到需求直接开始做了，结果产品和需求不一致
- 没有逻辑：写登录页面，不验证身份，这会导致啥问题，一个IP无限注册

那么如何保证需求的有效性和正确逻辑？

- 接到需求之后，自己先分析并记录对需求有疑问的点，把疑问点及时沟通
- 再次确认需求和实现方式，确认无误开始开发流程，在开发过程中也需要及时沟通

不要觉得弄明白了就直接上手写代码了，先计划再写代码

要是交付的代码于需求不对味，那后果是毁灭性的

分配时间的能力

开发实施：[软件设计](#) --> 思考最优的解决方案 --> 编码 --> 测试阶段 --> 完善 --> 交付

两项开发任务如何平衡自己的时间？

- 根据优先级安排开发任务，不能抉择时，谁负责谁做决定！

程序员很忙没时间 学习怎么办？

- 知识：事实性、概念性、程序性、反省认知
- 态度：接受、反应、评价、组织、内化
- 技能：记忆、理解、运用、分析、评价、创造

开发质量的能力

- 基本的判断标准，也就是经手开发的东西，能不能正常运行
- 代码的可读性高不高，别人读起来和修改维护的话，成本高不高？
- 模块化的程度，也就是拓展性怎么样？是否可以降低维护成本，提高开发效率？

解决问题的能力

项目或者系统总会或多或少遇到问题需要解决，特别是自己负责的模块。这就要看你对负责模块和系的熟悉程度，也考验深入分析问题和精准解决问题的能力

- 认清问题，就是先审题
- 寻找已有方案，就是看之前别人怎么解决的
- 解决问题，自己解决不了，就请教前辈

- 平时多积累经验，学习别人解决问题的思路 and 方式

持续学习的能力

互联网变化非常快，新技术层出不穷，要主动去接触新技术，而不是被迫的后知后觉

在没有足够的技术敏感性的时候，可以多关注前沿信息，开阔视野，了解最新技术解决了什么新问题是否能够给自己的工作带来帮助