下面这些题目任选其一(**运行平台均可自选Windows/Linux/Android**)：

|  |  |  |
| --- | --- | --- |
| 序号 | 题目 | 基本要求 |
| 1 | 文件搜索工具 | 仿照Windows的“搜索”功能或者“Everything”编写一个搜索工具。能够实时显示搜索结果并随时中断搜索。提供对文件属性的指定和过滤。  附加功能：正则表达式搜索功能和自动内码识别功能  难度：★★★★ |
| 2 | 多人点对点聊天软件 | 仿照QQ实现点对点聊天软件。可以同时与多人聊天，密码管理、有人上线自动提示，好友管理功能。能够显示表情、图片等。  附加功能：能够传送文件  难度：★★★ |
| 3 | 行计算器 | 编写一个拥有加减乘除取余以及三角函数、基本数学函数计算的行计算器（参考Java语言中的表达式）  如果输入出错，应指出错误原因。  注意是行计算器，写成windows中的计算器不能得分  难度：★★★ |
| 4 | 网速计算工具 | 编写一个UDP传输性能的测试工具，用以测试单向传输设备的性能。**设备环境**：默认两端计算机时间同步一致、计算机性能参数一致。两端计算机之间用单向传输设备连接（可以想象成连接在同一个家用路由器上）  **性能参数统计**：编写代码从发送端发送结构体到接收端，接收端解析结构体统计传输性能参数。结构体请自行设计，特别是第一个包中需要传送本次发送数据的总体情况。  最后得到传输性能参数，大致包含传输速率、总传输时间、平均每个包传输时间、丢包率、网络利用率、网络吞吐量等，如有其他更具代表性的参数更好。传输的数据最好是一个视频文件，因为大多数情况下，UDP的实际使用场景就是传输视频。  难度：★★★ |
| 5 | 函数画图工具 | 用户输入简单函数，程序自动画出函数图形  函数以y=f(x)的形式给出，其中f(x)可以是简单的四则运算，也可以是三角函数、幂函数、指数函数等，也可以是这些简单函数的复合。  可以仿照fooplot之类的工具来做  难度：★★★★ |
| 6 | 日程工具 | 仿照Android自带的日程工具做，一天可以设计多个日程可以在某一天设置1个或者多个活动日程，并存储该活动日程的时间、地点、时长等基本信息，并设置提醒功能  可选功能：每个日程允许重复（按日、按周、总次数、结束时间）  难度：★★★ |
| 7 | 音乐播放器 | 仿照千千静听、Foobar（或任意主流播放器）来做，除了完成音乐播放的基本功能外，还需提供音乐列表、歌词显示功能。歌词至少需滚屏，如果采用逐字渲染，将获得更高分数。可选功能：示波器、均衡器、音量增益等  难度：★★★ |
| 8 | 网站下载工具 | 对单个文件下载，需支持断点续传。支持一个页面中的多个链接依次下载。可显示下载进度，调整下载线程数目，调整扫描链接层数。对整个网站的批量下载后，支持在本地打开浏览。  难度：★★★★ |
| 9 | 接口测试工具 | 仿照PostMan来实现对Url接口的测试，可以用post或者get方法发送数据到url接口，获取其返回值，并解析给用户看。支持表单数据、Cookies，支持JSON解析  难度：★★★★ |
| 10 | 文件内码转换工具 | 将任意编码的纯文本文件转换成用户指定的编码文件。文件转换前后都要能打开查看。支持批量转换，支持指定编码和自动识别内码。  难度：★★★ |
| 11 | 简单Java开发工具 | 一个集成的Java开发环境，调用JDK来完成编译。需提供编辑界面、高亮度显示、编译环境和命令配置、编译错误显示、错误定位等基本功能。可参考vs code来实现。  可选功能：自动代码补全、调试。  难度：★★★★★ |
| 12 | 代理抓包分析工具 | 通过将自身设置为代理服务器，抓获其他应用程序的通讯包，并进行简单的分析，能够以16进制和字符串形式显示给用户查看。参考burp suite或者wire shark的基本功能来实现。  难度：★★★★★ |
| 13 | 文件批量改名工具 | 支持对本地文件的批量更改名称，可以使用各种通配符来实现改名目标，参考ReNamer来实现  难度：★★ |
| 14 | 连连看小游戏 | 参考小游戏连连看来实现  难度：★★★ |
| 15 | 贪吃蛇小游戏 | 参考贪吃蛇来实现  难度：★★ |
| 16 | 俄罗斯方块 | 参考传统俄罗斯方块来实现  难度：★★★ |
| 17 | 扫雷小游戏 | 参考windows中的扫雷来实现  难度：★★ |
| 18 | 五子棋 | 包括网络双人对战和人机对战模式，具有基本的游戏功能和简单AI  难度：★★★ |
| 19 | 中国象棋 | 只需要实现网络双人对战功能，按照中国象棋规则判断落子合法性、禁手、胜负平  难度：★★★★ |

注意：所有的源程序书写要规范，必要的地方要有注释，要有编译和使用说明文档。

一人一题，不能合作完成。

不能解决的问题，可以在网上查找资料，但切勿完整剽窃他人代码，一旦发现，记为0分

如果不想在上面的题目中选择，可以自行选题，但需事先告知老师，并获得老师同意。

评分规则：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 评分项 | 满分 | 80%-100% | 60%-79% | 40%-59% | 0%-39% |
| 选题 | 20分 | 题目难度很大，工作量饱满 | 题目难度较小，工作量饱满 | 题目没有多大难度，工作量一般 | 题目没有难度，工作量太少 |
| 算法正确性和效率 | 10分 | 通过所有测试数据，算法经过优化 | 不能正确处理某些刁难性质的测试数据，普通算法，未经过优化 | 对部分正常性质的测试数据不能正常处理，选择的算法效率低 | 大部分测试数据不能正确处理 |
| 程序编写风格 | 10分 | 自定义标识符命名规范，风格统一，见名知义；代码缩进对齐规范且风格统一；有清晰简明的注释，且格式规范。 | 自定义标识符命名基本规范，风格比较统一；代码缩进对齐基本规范；有适当的注释。 | 自定义标识符命名基本规范；代码缩进对齐基本规范，风格不统一；只有必要的注释。 | 自定义标识符命名比较随意，风格不统一；代码对齐不规范；只有极少量的或根本没有注释。 |
| 设计方案 | 10分 | 设计方案清晰合理，能综合考虑时间效率、空间效率、代码开发效率 | 设计方案比较合理，主要考虑了代码的开发效率，但时间、空间效率不够优化 | 设计方案尚可，但不能综合考虑各个影响因素，在某个方面有较大的缺陷。 | 没有明确的设计方案，完全不考虑各个因素的影响 |
| 程序稳定性 | 20分 | 对任意的非法输入都能处理，出错有提示，不会崩溃。 | 对大部分非法输入能正确处理，多数情况下有提示，极少数情况下会崩溃 | 对于大多数非法输入不能正确处理，只有少量提示，经常崩溃。 | 对于正常输入也不能正确处理，经常崩溃。 |
| 人机界面 | 10分 | 界面美观大方，用户操作简单快捷易上手，有明确的提示。 | 界面设计清楚，用户操作比较简单，有必要的提示 | 界面设计一般，缺乏美感，有些功能需要用户摸索，缺乏必要的提示。 | 人机交互体验很差，用户完全无法理解设计者的意图。 |
| 回答问题情况 | 20分 | 能正确回答所有问题 | 除了少数刁难性问题之外，能正确回答大部分问题 | 某些普通的问题也无法正确回答 | 大部分问题都不能正确回答 |