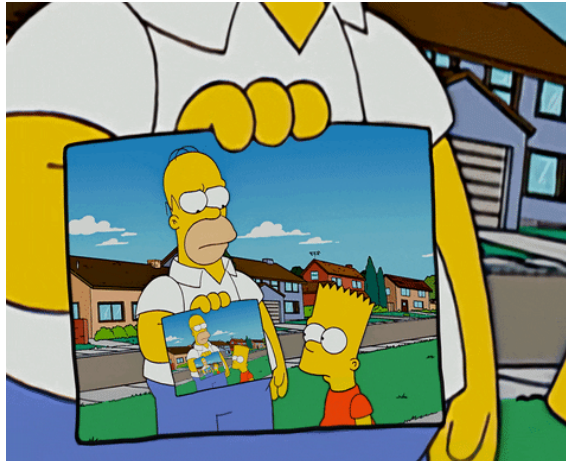


2019 年 软件工程实践 上机测试

——函数调用关系分析



From <https://media.giphy.com/media/3ov9jQX2Ow4bM5xxuM/source.gif>

考试要求

1. 考试时长 150 分钟。
2. 考试过程中可以查看自己携带的纸质或电子资料，但是不能上网。
3. 请使用 C++ 和 Qt 完成题目，可以使用 STL 和 Qt 中的各种库函数。
4. 请将整个 Qt 工程目录，在执行清理之后用 ZIP 打包上传，单个 ZIP 文件不大于 1MB。
5. 命名规则为：“学号+姓名.zip”（中间为加号）。请严格使用此命名规范。

一、 题目介绍

函数调用关系图（Call Graph）对于程序中控制流和数据流的分析具有重要作用。其内容是以图的方式，将函数与其函数体中调用的函数进行关联。

在本题目中，你需要对一个给定的源代码文件进行函数调用分析，并输出一个简化版的函数调用图。在给定的源代码文件中包含若干函数定义，一个函数的实现过程中可能会调用其他函数。在函数调用关系图中，需要从 `main` 函数开始，逐级展示被调用的函数。

二、具体要求

源文件内容约束：

1. 函数名不能作为变量，函数名和变量名均不会与 `func` , `call` , `end` 等关键字冲突。
2. 函数定义形式如下：

```
func 函数名
... 函数体
end
```

具体来说，函数定义由若干行组成：

第一行以 `func` 四个字母开头，随后为一个空格，再之后为函数名，函数名仅由英文字母、数字、下划线组成，本次测试中你无需检查函数名的正确性；

随后若干行为函数体；

函数定义的最后一行，由 `end` 三个字母组成。函数体中不会出现该行。

函数定义中不会再次出现新的函数定义，即函数定义不会嵌套。

3. 函数调用方法如下：

```
call 函数名
```

每个函数调用占一行，以 `call` 四个字母开头，随后为一个空格，再之后为需要调用的函数名。你无需检查函数名的正确性，测试中保证函数名满足命名规范，且该函数在代码中有定义，但请注意，在源代码中，函数定义可能出现在函数调用之后。

4. 函数不会作为函数调用的参数，不支持函数指针。
5. 函数调用过程中不会出现环（因此不会有递归调用）。
6. 代码中一定会有一个 `main` 函数被定义。
7. 测试文件中的单行代码长度不超过 999 个字符（不包括结尾的换行字符）。
8. 在本次测试中，你可以仅关注函数定义和函数调用，其他部分的代码无需处理。
9. 代码中不会出现循环（`FOR`、`WHILE` 等）和条件判断（`IF` 等）。

函数调用关系输出格式：

输出的第一行包含 `main` 四个字母，表示从 `main` 函数开始分析。

此后按照调用的顺序深度优先依次打印 `main` 中调用到的函数。每一次调用需要将函数名缩进一层。缩进格式为：一个 “|” 符号后接三个空格。每次函数返回时取消一层缩进。（可参照后面的示例图）

因此打印出的结果，应满足以下性质：

1. 第一行只包含 `main`；
2. 被 `main` 直接调用的函数名在缩进第一层打印；
3. 被 `main` 经过 n 个函数调用后被调用的函数名在缩进第 $n+1$ 打印。

GUI 要求：

1. 按照如下截图绘制/创建用户界面。
2. 当用户点击“选择文件”按钮，应弹出文件选择框。在用户选定文件之后，将文件内容显示在“原文件内容”下的文本框中，并清空“函数调研关系”下的文本框。
3. 随后，当用户点击“分析调用”时，程序将从 `main` 函数开始，分析源文件中的函数调用情况，并将结果展示在“函数调用关系”下的文本框中。
4. 若用户未选择文件，直接点击了“分析调用”，则不进行任何操作。
5. 其他未定义的用户行为不作要求（可随意处理），但请保证程序不会崩溃。



评分标准

1. 能够正确绘制/创建出 UI 界面（20 分）
2. UI 界面能够响应用户交互（25 分）
3. 能够成功读取并展示出源文件内容（20 分）
4. 能够展示出正确的函数调用关系（30 分）（其中内容和格式各 15 分）
5. 代码简洁易读、风格统一（5 分）

一些提示

1. 可使用 `std::map`、`QMap`、`std::vector` 等结构保存信息；
2. 如果使用 `std::string`，可使用其 `compare` 函数来比较子字符串；
3. 在递归中，可以将递归层数通过 `int` 或者 `string` 类型参数进行传递，以决定在输出时，应先打印多少次缩进标志。