

编译原理研讨课LLVM实验PR001任务书

熟悉Clang的安装和使用

实验内容：

本部分主要包含两个内容：

1. 掌握如何从源代码编译安装LLVM和Clang
2. 了解如何生成和查看C程序对应的AST

验收标准：

1. 正确编译生成Clang和LLVM的可执行文件
2. 可以查看一个C程序对应的AST

添加overflow check操作的制导

实验内容：

在程序中添加overflow check操作的制导，区域以函数/过程的定义(不是声明)为单位，形如：

```
#pragma overflowCheck
int func_name(){
    .....
}
```

对于添加了制导的源程序*.c，按照规则在编译时打印每个函数的名称，该函数是否在制导范围内。

对于一个函数的是否在制导范围内的定义：

1. 一个制导总是匹配在其后出现的，离它最近的一个函数定义
2. 一个制导只能匹配最多一个函数定义

说明：

1. 制导和匹配的函数定义间可以有任何语句，没有限制
2. 打印函数名称功能无需自行从头实现，课程将提供模板供参考，见“输出插件使用说明”

示例：

Sample code 1:

```
void foo1(){
    ....
}
```

Sample output: "foo1: 0"

Sample code 2:

```
#pragma overflowCheck
void foo2(){
    ....
}
```

Sample output: "foo2: 1"

验收标准：

1. 实验报告提交到课程网站
 1. 课程网站有模板供参考。
2. 实验源代码提交至每组对应的Gitlab账号的LLVM工程当中(master分支)，我们将重新编译各位同学的代码，并根据相应的case进行检查。
 1. `http://124.16.71.65/llvm1/llvm.git`，其中 `llvm1` 是每组的用户名，当在课程提供的服务器上操作时。密码为 `llvm1!llvm1!`，每组根据自己组名调整。