

竞赛要求说明

题目要求

要求使用 **C或C++** 实现一个编译器，将 **SysY2020语言** 的代码翻译为 **Arm汇编代码**。

评测流程简单说明

参赛队提交的编译器代码将首先在X86 CentOS的服务器上被编译运行，将测试样例的SysY2020语言的代码翻译为Arm汇编代码，**上述操作在X86服务器中进行**。之后再将生成的Arm汇编代码传入**树莓派**中进行汇编、链接并运行，并依据运行结果进行评测。

大赛指定的编译环境主要参数

大赛指定的编译环境主要参数如下：

X86服务器

1. Docker容器操作系统：Ubuntu18.04
2. 编译器：LLVM/Clang-8.0，编译选项
C语言：clang -std=c11 -O2 -lm
C++：clang++ -std=c++17 -O2 -lm

Raspberry Pi 4 Model B Rev 1.1

1. CPU：Broadcom BCM2835, Quad core Cortex-A72 (ARM v7 rev 3) 64位SoC，主频1.5GHz。
2. 内存：2GB LPDDR4-3200 SDRAM。
3. GPU：500Mhz VideoCore VI。
4. 操作系统：Raspbian GNU/Linux 10 (buster)，基于Debian。（树莓派的操作系统是**32位**的）
5. 编译器：gcc version 7.5.0 (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04)

评测环境说明

平台的评测环境使用 **LLVM/Clang-8.0** 编译环境，请自行测试在此环境下能否成功编译和运行。鉴于不同的编译环境下运行结果可能存在较大的差异，可在代码实现过程中在平台上多提交几次，观察能否在测试环境下成功编译。

实现的编译器要求

1. 提交的编译器应支持指定的**SysY2020语言**。关于SysY2020语言的定义请参考compiler.educg.net上提供的详细说明文件"20191228-SysY语言定义.pdf"。
2. 提交的编译器需要具有 **词法分析**、**语法分析**、**语义分析**、**目标代码生成与优化** 等功能。
3. 对于正确编译通过的SysY2020基准测试程序，应生成符合要求的 **ARM汇编文件**，要求能被编译链接成可执行文件，并在安装有Linux操作系统的指定ARM硬件平台上加载并运行。生成的汇编程序应为**32位**，**ARM v7**体系结构。
4. 生成的ARM汇编文件应首先保证正确性，再依照程序执行的性能来评分。
5. 参赛队需从头构造编译系统，允许基于**Lex**、**Yacc**、**Bison**等通用词法、语法解析器生成工具帮助生成部分代码，但不得直接使用GCC、LLVM等现有、开源编译器及框架的源代码及裁剪。

需提交的文件

1. 综合编译系统设计的完整工程文件（必须包含全部C/C++源代码、Make文件等），并在竞赛平台上至少有一次完整通过性能测试的记录和有效成绩。
2. 综合编译系统统设计文档。

语言标准

大赛仅可使用C或C++实现编译器。

- C语言：应符合ISO/IEC 9899: 2011即C11 标准；
- C++：应符合C++17标准。