

# YAME 使用手册

## 一、YAME 简介

欢迎您使用 YAME，一个简洁易用的 MIPS 汇编器、反汇编器和模拟器。YAME 是 Yet Another MIPS Editor 的缩写，也可以认为是 YAME: A MIPS Editor 的缩写。YAME 使用 Python 语言开发，并使用 PyQt5 作为图形界面库进行图形界面的设计。以下将为您介绍 YAME 的特性以及使用方法。

## 二、YAME 特性

YAME 作为一个简洁易用的 MIPS 开发工具，最大的亮点就是对众多 MIPS 指令的支持，以及自定义伪指令的功能。另外，YAME 还具有一个能够模拟终端的模拟器，支持部分 syscall 指令，可以说功能齐全。

### 2.1、YAME 支持的指令

YAME 支持以下 MIPS 指令的汇编、反汇编与模拟：

#### R 指令

- |        |         |           |
|--------|---------|-----------|
| • add  | • mthi  | • sltu    |
| • addu | • mtlo  | • sra     |
| • and  | • mult  | • srav    |
| • div  | • multu | • srl     |
| • divu | • nor   | • srlv    |
| • jalr | • or    | • sub     |
| • jr   | • sll   | • subu    |
| • mfhi | • slv   | • syscall |
| • mflo | • slt   | • xor     |

#### I 指令

- |          |          |         |
|----------|----------|---------|
| • addi   | • bltz   | • lw    |
| • addiu  | • bltzal | • ori   |
| • andi   | • bne    | • sb    |
| • beq    | • lb     | • slti  |
| • bgez   | • lbu    | • sltiu |
| • bgezal | • lh     | • sh    |
| • bgtz   | • lhu    | • sw    |
| • blez   | • lui    | • xori  |

#### J 指令

- |     |       |
|-----|-------|
| • j | • jal |
|-----|-------|

### 格式指令

- |          |           |          |
|----------|-----------|----------|
| • .text  | • .8byte  | • .dword |
| • .data  | • .ascii  | • .half  |
| • .2byte | • .asciiz | • .space |
| • .4byte | • .byte   | • .word  |

### syscall 编号

- |             |               |                 |
|-------------|---------------|-----------------|
| • 1 (输出整数)  | • 10 (退出)     | • 41 (随机整数)     |
| • 4 (输出字符串) | • 11 (输出字符)   | • 42 (有范围的随机整数) |
| • 5 (输入整数)  | • 12 (读入字符)   |                 |
| • 8 (输入字符串) | • 30 (读入系统时间) |                 |

## 2.2、自定义伪指令

为了方便编程人员，减少工作量，YAME 提供自定义伪指令的功能。

您只需要输入自定义伪指令的名称、操作数数量，并遵循我们的伪指令格式，即可自定义伪指令。详细的操作方式请见。

## 2.3、带有终端的模拟界面

YAME 提供一个带有终端的模拟界面，可以模仿 syscall 的各种输入输出指令。

模拟界面的详细操作方式请见。

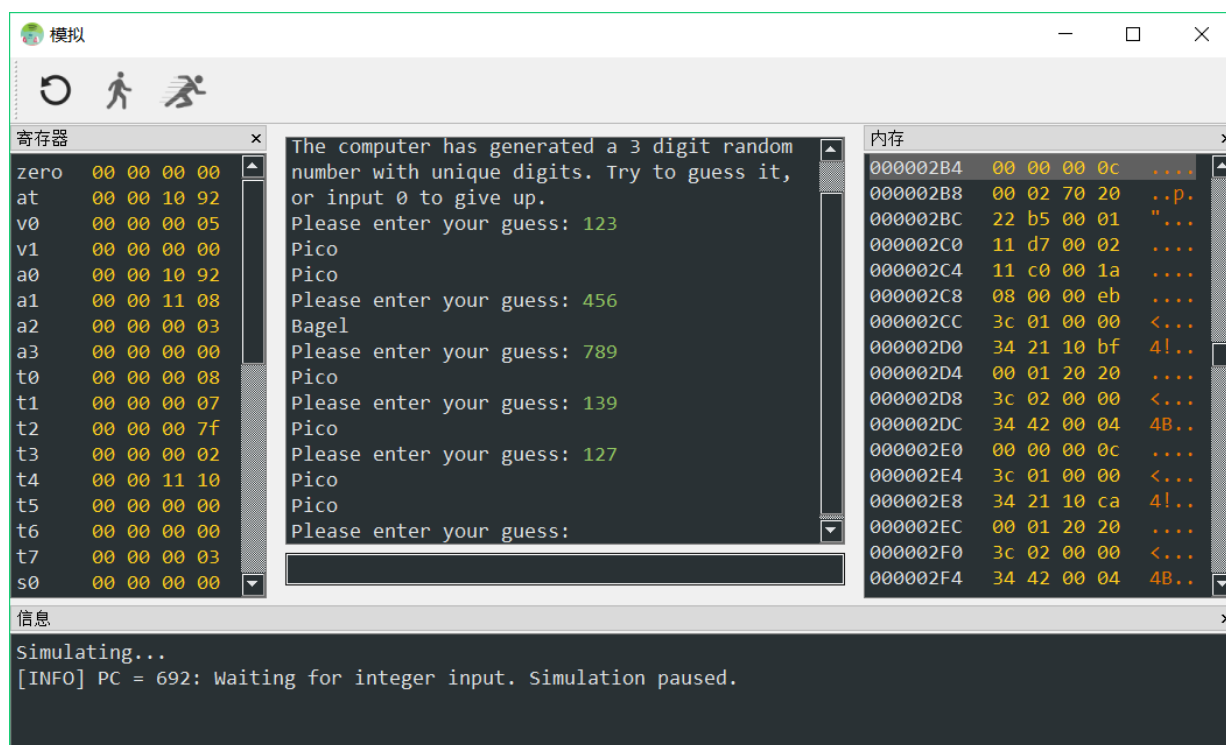


图 2.1 – 带有终端的模拟界面

### 三、操作指南

本章节将会详细介绍 YAME 各个功能的使用方式。

#### 3.1、界面

本节介绍 YAME 的界面。

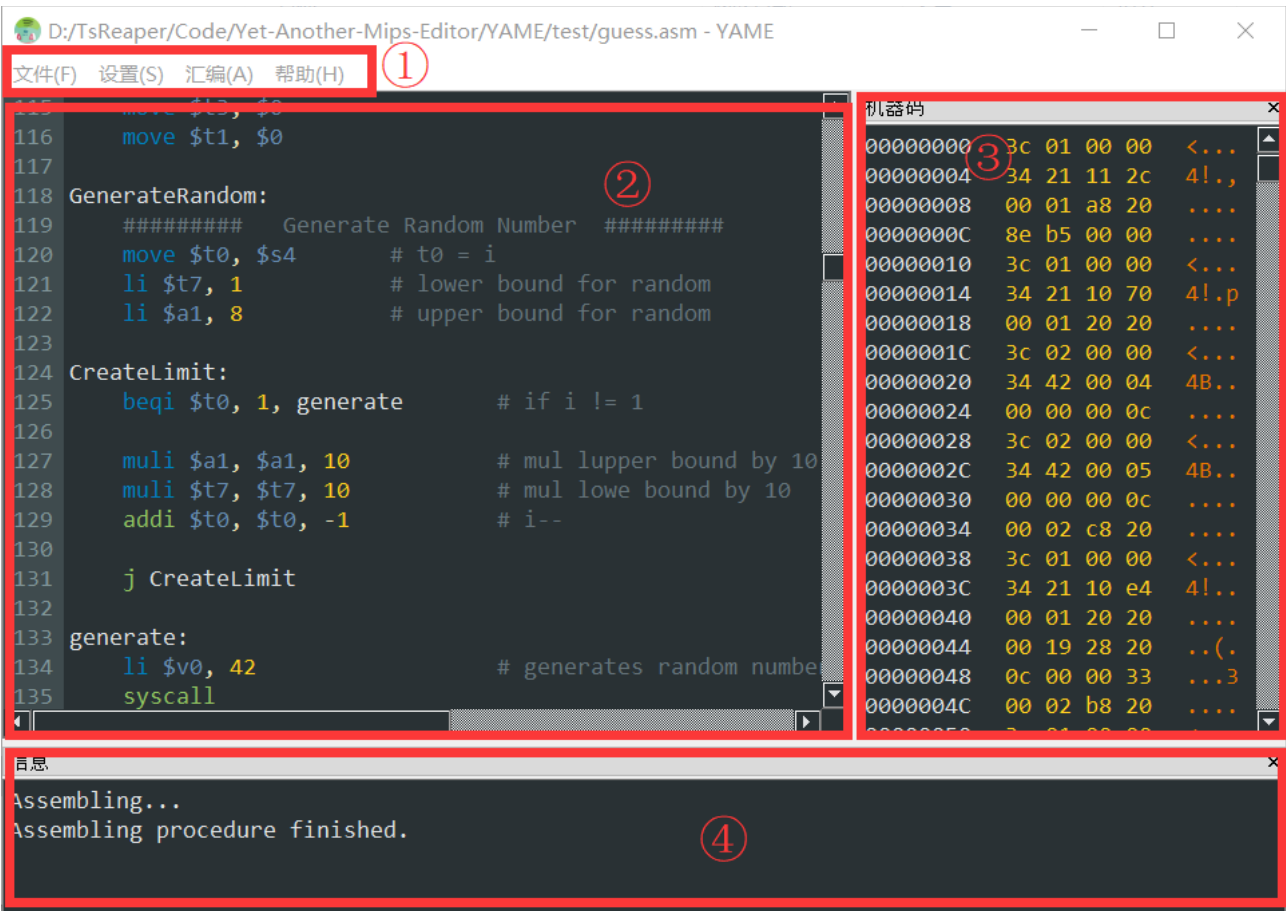


图 3.1 – 编辑界面

图 3.1 展示的是 YAME 的编辑界面。该界面主要进行汇编代码的编辑、汇编与反汇编。

- 区域 ① 是菜单栏，在这里调用 YAME 的各种功能；
- 区域 ② 是编辑区域，在这里编辑 MIPS 代码；
- 区域 ③ 是机器码区域，在这里显示 MIPS 代码汇编后的结果；
- 区域 ④ 是信息区域，在这里显示汇编 / 反汇编过程中产生的各种提示。

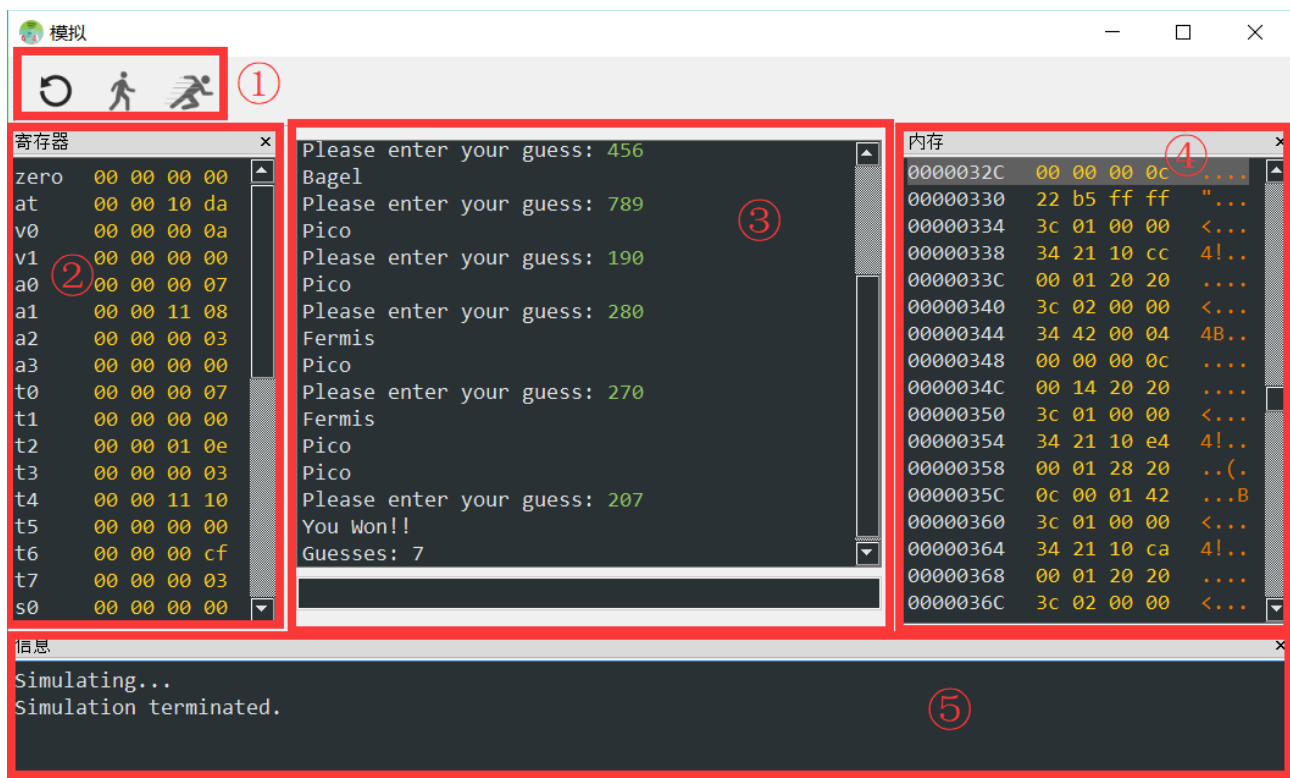


图 3.2 – 模拟界面

图 3.2 展示的是 YAME 的模拟界面。该界面主要进行 MIPS 代码的模拟运行。

区域 ① 是菜单栏，用于进行模拟器重置、单步运行与全部运行；

区域 ② 是寄存器区域，显示当前模拟器中各寄存器的值；

区域 ③ 是终端区域，用于显示并接受送至 MIPS 程序的数据（由 syscall 产生）。

区域 ④ 是内存区域，显示当前模拟器中内存的值；

区域 ⑤ 是信息区域，显示模拟过程中产生的各种提示。

## 3.2、菜单栏

本节介绍位于界面最上方的菜单栏。

文件菜单中的功能如下：

- 新建：清空当前窗口的内容，新建一个空的文件；
- 打开：打开硬盘上的汇编（.asm）代码文件；
- 保存：将当前窗口的编辑内容保存到磁盘；
- 另存为：将当前窗口的编辑内容保存到磁盘的其它位置；
- 导入：导入磁盘上的 .coe 文件或二进制（.bin）文件，文件导入后会自动反编译，结果显示在编辑区域内；
- 导出：将编辑区域中的汇编代码自动汇编后，导出为 .coe 文件或二进制（.bin）文件；

- 退出：关闭 YAME。

设置菜单中的功能如下：

- 汇编器设置：打开汇编器设置窗口；
- 伪指令：打开自定义伪指令的窗口。

汇编菜单中的功能如下：

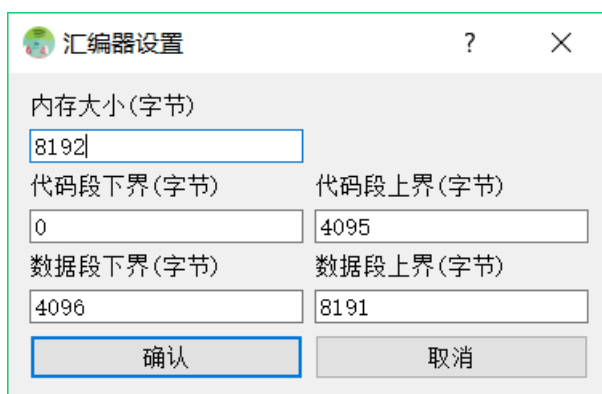
- 汇编：将编辑区域中的汇编代码进行汇编；
- 反汇编：将机器码区域中的机器码进行反汇编；
- 模拟：将编辑区域中的汇编代码进行模拟。

帮助菜单中的功能如下：

- 关于：显示 YAME 的有关信息。

### 3.3、汇编器设置

本节介绍汇编器设置界面。在这个界面中可以对汇编器进行设置。



该对话框标题为“汇编器设置”，包含以下输入项：

内存大小(字节)	
8192	

代码段下界(字节)	代码段上界(字节)
0	4095

数据段下界(字节)	数据段上界(字节)
4096	8191

底部有两个按钮：“确认”和“取消”。

图 3.3 – 汇编器设置界面

YAME 汇编器的代码段范围与数据段范围需要用户自行设置。该设置会影响到汇编结果与反汇编结果，所以需要设置正确。

如果填入的五个数据互相冲突，点击确认后将会报错，否则将成功保存。

### 3.4、自定义伪指令

本节介绍伪指令界面。在这个界面中可以自行添加伪指令。



图 3.4 – 伪指令设置界面

点击“添加”按钮新建一条伪指令，在“伪指令名”中填入伪指令的名称，在“操作数数量”中填入伪指令的操作数数量，在“真指令”中填入伪指令对应的真指令，再点击“保存”按钮即可保存伪指令。伪指令会在汇编的过程中被自动替换为真指令。

在真指令中，如果需要使用伪指令的操作数，可以用 [x] 代指第 x 个操作数。另外，可以用 @ 标签名 @ 获取标签所在的地址。该语法在汇编器的编辑界面中也是有效的。

如果想要删除一条伪指令，只需要在左边的列表中选中该伪指令，并点击“删除”按钮即可。

如果想要修改一条伪指令，只需要在左边的列表中选中该伪指令，完成修改操作后点击“保存”按钮即可。

### 3.5、模拟

本节介绍 YAME 的模拟功能。在编辑界面按下 F11 键，或通过菜单栏中的“汇编 – 模拟”即可打开模拟器。

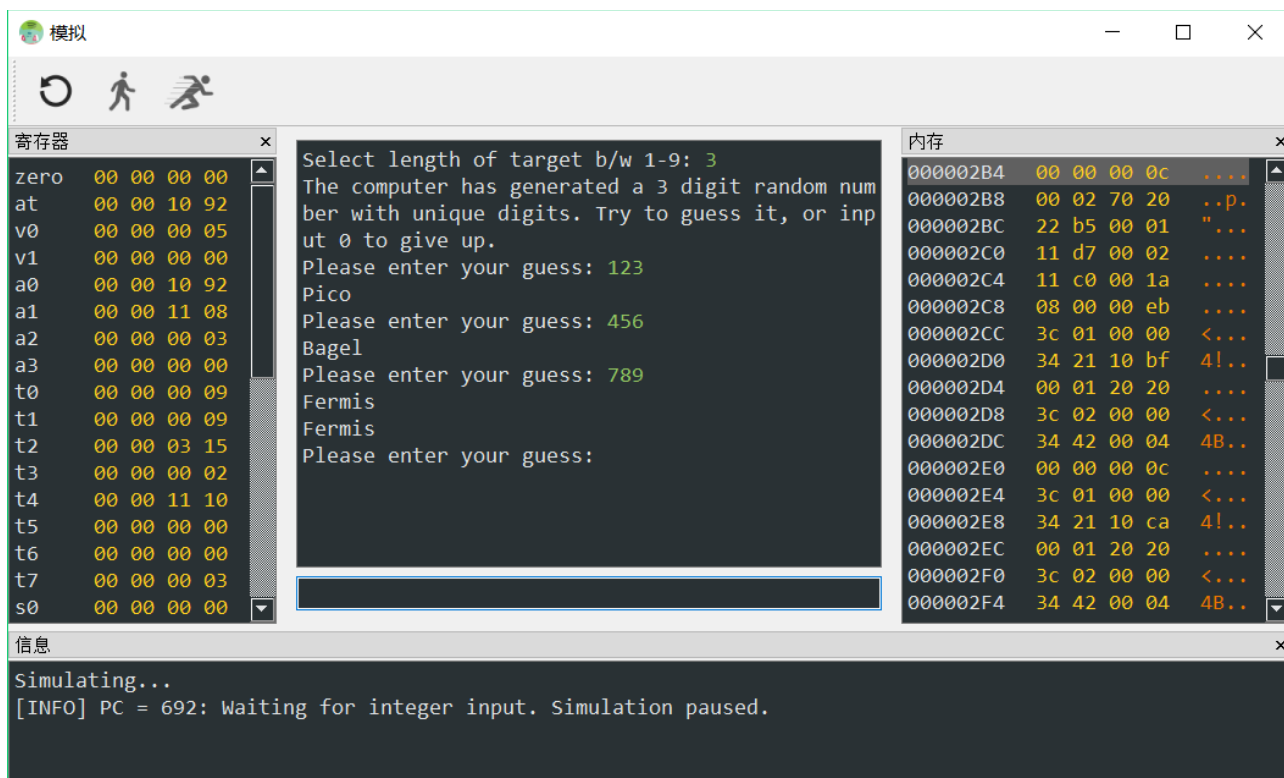


图 3.5 – 模拟界面

在模拟界面中，可以按下 F9 键，或点击菜单栏中的“圆箭头”按钮重置模拟器；可以按下 F10 键，或点击菜单栏中的“走路”按钮单步执行 MIPS 程序；可以按下 F11 键，或点击菜单栏中的“跑步”按钮全部执行 MIPS 程序。

界面两边的寄存器与内存区域会显示当前模拟器的寄存器与内存的值，便于用户调试。

界面中间的终端会显示 MIPS 程序需要显示的数字或字符串。在 MIPS 程序需要用户输入时，下方的输入框将会激活，用户可以输入数据并按下回车键以给 MIPS 程序输入数据。在终端中，MIPS 程序输出的数据以白色字体表示，用户输入的数据以绿色字体表示。