

# 计算机组成课程 Mars 使用说明

## 启动 Mars

Windows 系统（已安装 Java 环境）双击 jar 包即可正常使用。

Linux 系统可以使用配套的启动脚本 `Mars.sh`，双击脚本文件即可打开 Mars。Mars 的文件名应为 `Mars.jar` 且和 `Mars.sh` 应位于同一目录。如双击脚本弹出文本编辑器而没有打开 Mars，则需先用终端执行 `chmod +x Mars.sh`。

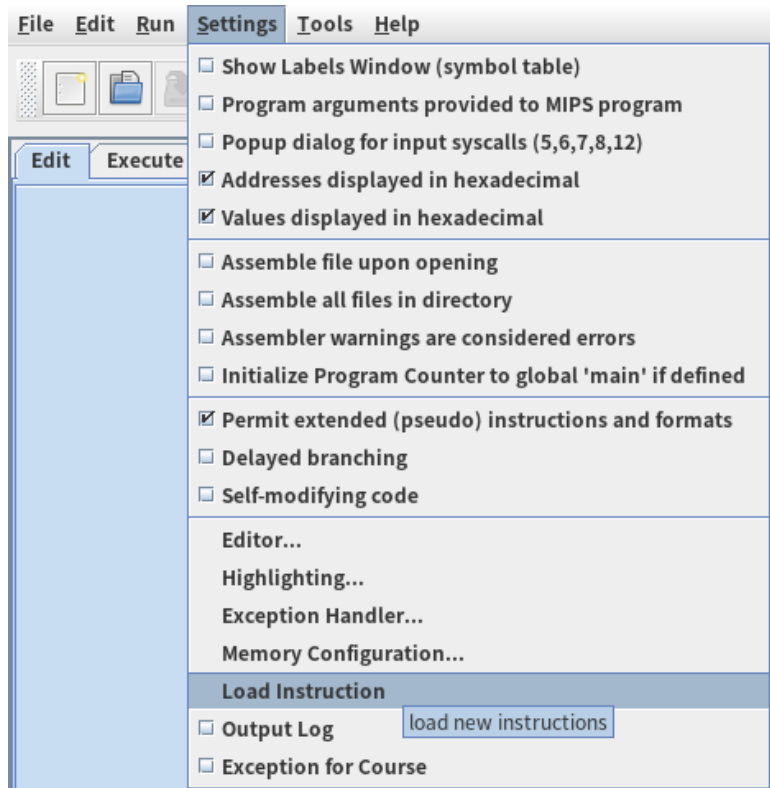
## GUI 操作

### 加载新指令

利用该 Mars 加载含有新增指令信息的 `.class` 文件，即可对新指令进行测试。

**注意：加载的 `.class` 文件名必须与指令名相同且字母全部为小写！**

1. 点击 `Settings` 菜单中的 `Load Instruction` 选项。



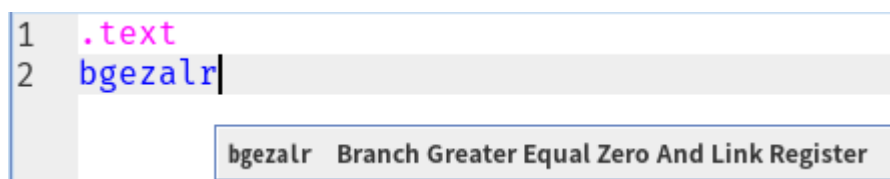
2. 选择要导入的新指令 `.class` 文件（通常可在题目下方下载），并点击 打开。



3. 弹出成功提示，加载指令完成。



4. 加载指令后即可编写含有课上新指令的测试程序。

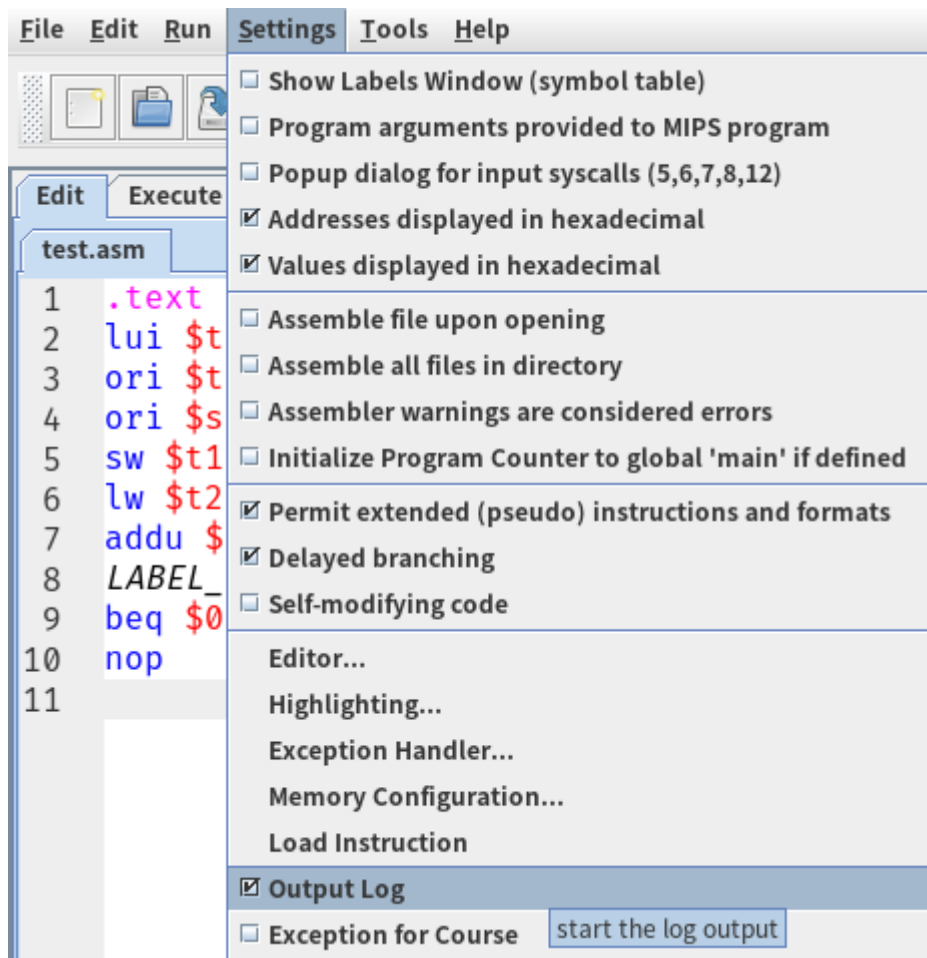


## 输出运行信息

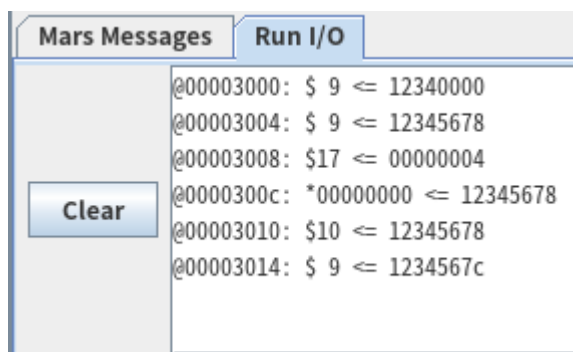
该 Mars 可以在运行汇编程序时输出模拟 CPU 运行信息：

- 写寄存器： `@%08h: $%2d <= %08h`
- 写入内存： `@%08h: *%08h <= %08h`

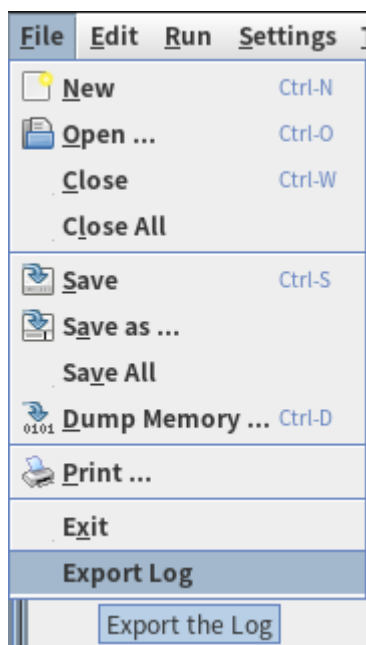
1. 点击 `Settings` 菜单，勾选 `Output Log` 选项。



2. CPU 运行信息将被输出到界面下方的 **Run I/O** 窗口中。



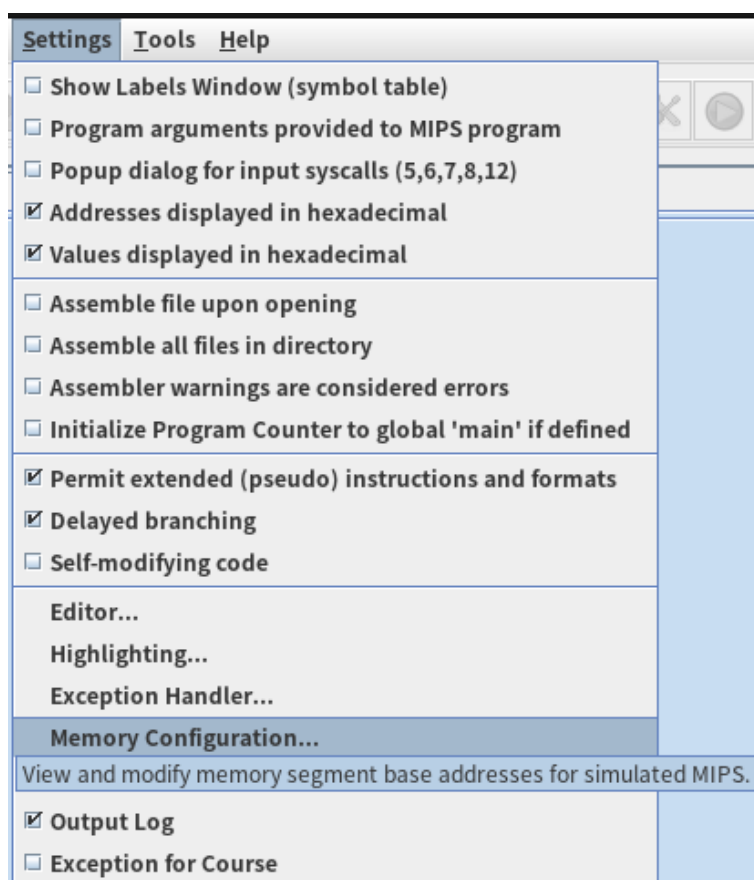
3. 点击 **File** 菜单的 **Export Log** 选项，可将输出的运行信息导出至文本文件中。



## 支持新 IM 地址空间

原版 Mars 中 Compact, Data At Address 0 地址空间不支持程序超出 0x3000-0x3ffc 范围, 在该版本 Mars 中可以选择 .text 范围为 0x3000-0x6ffc 的地址空间, 从而编写更长的测试程序。

1. 在 Settings 菜单中选择 Memory Configuration 。



2. 选择 Compact, large Text 地址空间, 并点击 Apply and Close 。



3. 在该模式下可编写长度更长, 指令地址超过 0x4000 的测试程序。

Edit Execute				
Text Segment				
Bkpt	Address	Code	Basic	
<input type="checkbox"/>	0x00003ff0	0x340903fd	ori \$9,\$0,0x000003fd	1021: ori \$t1, \$0, 1021
<input type="checkbox"/>	0x00003ff4	0x340903fe	ori \$9,\$0,0x000003fe	1022: ori \$t1, \$0, 1022
<input type="checkbox"/>	0x00003ff8	0x340903ff	ori \$9,\$0,0x000003ff	1023: ori \$t1, \$0, 1023
<input type="checkbox"/>	0x00003ffc	0x34090400	ori \$9,\$0,0x00000400	1024: ori \$t1, \$0, 1024
<input checked="" type="checkbox"/>	0x00004000	0x34090401	ori \$9,\$0,0x00000401	1025: ori \$t1, \$0, 1025
<input type="checkbox"/>	0x00004004	0x34090402	ori \$9,\$0,0x00000402	1026: ori \$t1, \$0, 1026
<input type="checkbox"/>	0x00004008	0x34090403	ori \$9,\$0,0x00000403	1027: ori \$t1, \$0, 1027
<input type="checkbox"/>	0x0000400c	0x34090404	ori \$9,\$0,0x00000404	1028: ori \$t1, \$0, 1028
<input type="checkbox"/>	0x00004010	0x34090405	ori \$9,\$0,0x00000405	1029: ori \$t1, \$0, 1029
<input type="checkbox"/>	0x00004014	0x34090406	ori \$9,\$0,0x00000406	1030: ori \$t1, \$0, 1030
<input type="checkbox"/>	0x00004018	0x34090407	ori \$9,\$0,0x00000407	1031: ori \$t1, \$0, 1031
<input type="checkbox"/>	0x0000401c	0x34090408	ori \$9,\$0,0x00000408	1032: ori \$t1, \$0, 1032
<input type="checkbox"/>	0x00004020	0x34090409	ori \$9,\$0,0x00000409	1033: ori \$t1, \$0, 1033
<input type="checkbox"/>	0x00004024	0x3409040a	ori \$9,\$0,0x0000040a	1034: ori \$t1, \$0, 1034

# 命令行操作

---

## 加载新指令

命令中加入 `cl` 选项后跟新指令 `.class` 文件路径以运行新指令测试程序。

## 输出运行信息

命令行加入 `lg` 选项可以向控制台标准输出打印 CPU 运行信息。

## 使用新地址空间

命令中使用 `mc LargeText` 选项。

## 参考示例

测试新指令 `bgezalr`，测试程序为 `test.asm`：

```
java -jar Mars.jar nc lg db mc LargeText cl bgezalr.class 200000 test.asm
```

其中 `nc` 表示不显示 Mars 版权信息，`lg` 表示开启 CPU 输出信息，`db` 表示开启延迟槽，`mc LargeText` 表示使用新地址空间，`cl bgezalr.class` 表示载入新指令文件 `bgezalr.class`，`200000` 表示限制指令执行条数，达到该值时停止执行（具体值可自定，能使得 Mars 在测试程序以无限循环结尾时能运行结束即可）。

执行上述命令时，需保证 `Mars.jar`，`bgezalr.class`，`test.asm` 三个文件位于相同目录，且执行命令的终端也处于这个目录。

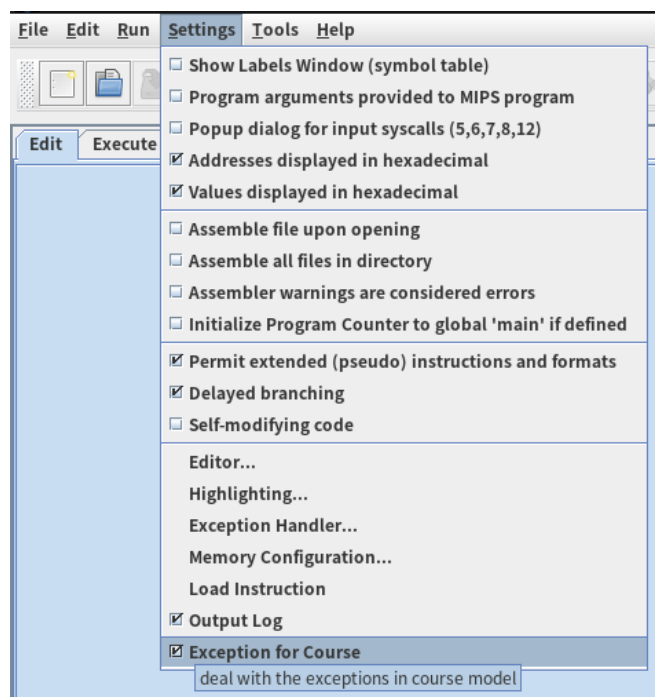
正确执行后，控制台标准输出将显示测试程序的运行信息（以下输出内容仅供参考）：

```
@00003000:$ 9 <= 00000001
@00003004:$10 <= 00003014
@00003008:$11 <= 00003010
@0000300c:$11 <= 00003014
@00003014:*00000004 <= 00000001
@00003018:$ 9 <= 00005679
```

# 中断异常说明

本 Mars 支持课程 P7 要求的异常和定时器中断，开启方法：

- 对于 GUI 操作：勾选 Settings 菜单中的 Exception for Course 选项。



- 对于命令行：添加 `ex` 参数以使用 P7 的中断异常功能。

示例：`java -jar Mars.jar nc db lg ex mc LargeText 100000 test.asm`

在该模式下 Mars 支持 P7 要求的两个定时器：

- 在右侧寄存器显示区域选择 `timer1` 和 `timer2` 查看定时器中三个寄存器的状态。

Registers	Coproc 1	Coproc 0	timer1	timer2
Name		Number	Value	
\$0 (ctrl)		0	0x00000000	
\$1 (preset)		1	0x00000000	
\$2 (count)		2	0x00000000	

指令行为的调整：

- 在该模式下 `add`、`sub`、`addi` 指令会检测溢出并产生异常，否则不会检测。
- 在该模式下 `syscall` 不会进行输入输出操作，会直接产生异常。