HW1

编译选项含义

- -E, --preprocess
  - Only run the preprocess
- -S, --assemble
  - Only run preprocess and compilation steps
- -c, --compile
  - Only run preprocess and compile and assemble steps
- -o<file>, --output <arg>, --output=<arg>
  - Write output to <file>
- -m32/64

预处理器

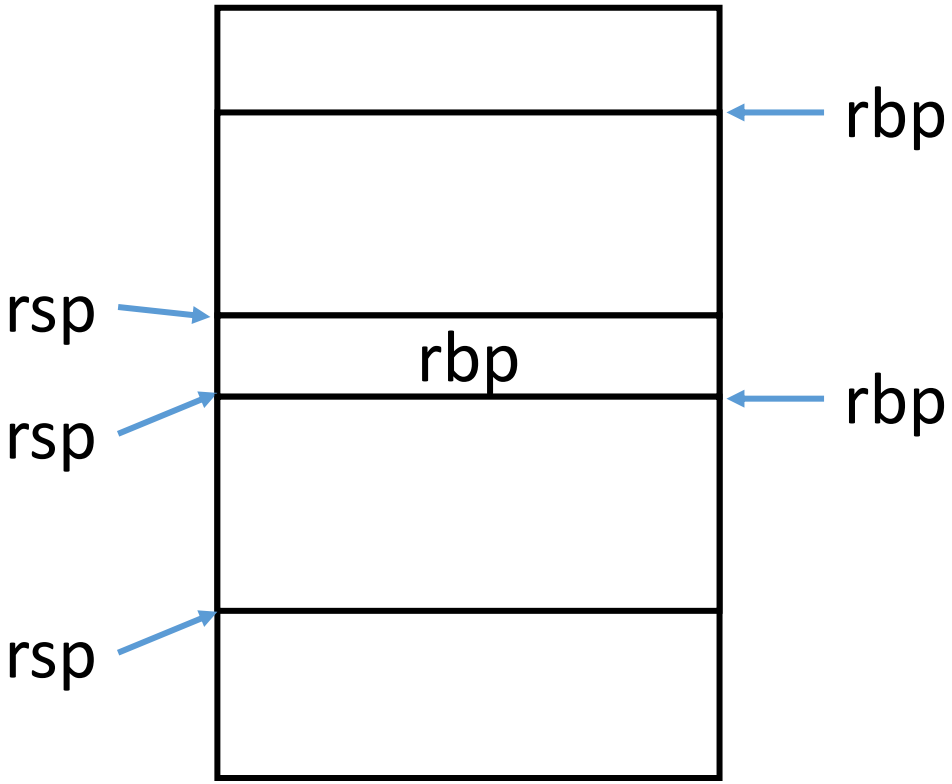
- Header files
- Macros
- Conditionals
- Diagnostics
- Line Control
- Pragmas

```
#ifdef __vax__
#error "Won't work on VAXen. See comments at
get_last_object.“
#endif

#if !defined(FOO) && defined(BAR)
#error "BAR requires FOO."
#endif
```

源文件 & 汇编文件

栈帧的创建与撤销




```c
#include "stdio.h"

int main(int argc, char const *argv[])
{
  printf("hello world\n");
  return 0;
}
```

```
main:                                   # @main
  .cfi_startproc
# %bb.0:
  pushq %rbp
  .cfi_def_cfa_offset 16
  .cfi_offset %rbp, -16
  movq  %rsp, %rbp
  .cfi_def_cfa_register %rbp
  subq  $32, %rsp
  movabsq $.L.str, %rax
  movl  $0, -4(%rbp)
  movl  %edi, -8(%rbp)
  movq  %rsi, -16(%rbp)
  movq  %rax, %rdi
  movb  $0, %al
  callq printf
  xorl  %ecx, %ecx
  movl  %eax, -20(%rbp)         # 4-byte Spill
  movl  %ecx, %eax
  addq  $32, %rsp
  popq  %rbp
  retq
.Lfunc_end0:
  .size main, .Lfunc_end0-main
  .cfi_endproc
```

源文件 & 汇编文件

Calling convention

- 参数寄存器
  - RDI, RSI, RDX, RCX, R8, R9
- 返回值寄存器
  - RAX
- 函数调用前调用者保存参数寄存器到栈帧中

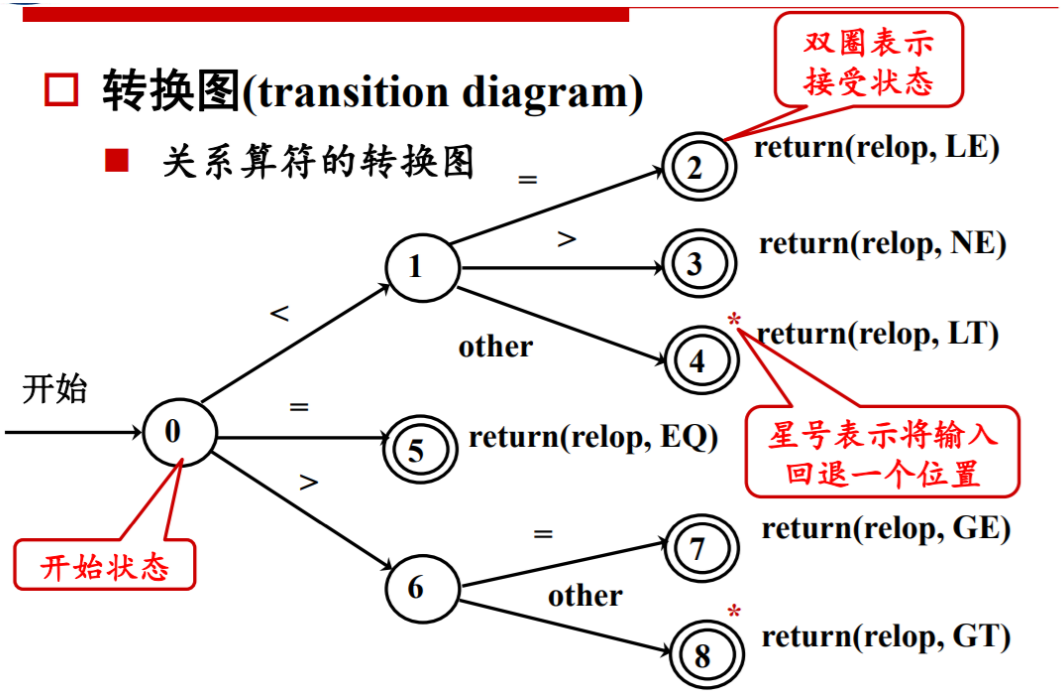
```c
#include "stdio.h"

int main(int argc, char const *argv[])
{
  printf("hello world\n");
  return 0;
}
```

```
main:                                   # @main
  .cfi_startproc
# %bb.0:
  pushq %rbp
  .cfi_def_cfa_offset 16
  .cfi_offset %rbp, -16
  movq  %rsp, %rbp
  .cfi_def_cfa_register %rbp
  subq  $32, %rsp
  movabsq $.L.str, %rax
  movl  $0, -4(%rbp)
  movl  %edi, -8(%rbp)
  movq  %rsi, -16(%rbp)
  movq  %rax, %rdi
  movb  $0, %al
  callq printf
  xorl  %ecx, %ecx
  movl  %eax, -20(%rbp)         # 4-byte Spill
  movl  %ecx, %eax
  addq  $32, %rsp
  popq  %rbp
  retq
.Lfunc_end0:
  .size main, .Lfunc_end0-main
  .cfi_endproc
```

词法分析程序

- 把右图编码为一张二维状态转换表，然后根据输入查表转换状态即可，需要注意状态4和8的回退操作
- 识别过程
  - <==><，01205068014
  - a<=b，other 012 other
- Makefile
  - Makefile中要提供clean伪目标，用于清理生成的文件



Answer.md

- 尽快熟悉markdown的语法规则
- 涉及到代码解释时，不要把整个源文件拷贝到answer.md文件中，截取代码片段，能将问题解释清楚即可
- 适当的使用粗体和斜体，通篇粗体或斜体起不到强调作用
- 仔细读题，不要遗漏问题

实验提交

- git commit的描述信息要简洁明确的概括本次提交内容与目的
- 避免提交无关文件
  - 使用.gitignore
  - 提交前make clean清除编译产生的中间文件
  - 提交前git status检查暂存了哪些文件，剔除无关文件