



考点	重要程度	占分	题型
1.指令类型	★★★★★	5~10	选择 填空
2.MIPS源程序框架	必考		大题
3.系统调用syscall			

考点1 指令类型

R-type	OP(6)	RS1(5)	RS2(5)	RD(5)	SA(5)	OPX(6)
--------	-------	--------	--------	-------	-------	--------

I-type	OP(6)	RS(5)	RD(5)	Immediate
--------	-------	-------	-------	-----------

条件分支跳转指令(b): 编码语的后16位broffset是相对PC的有符号偏移量, 由于指令是4字节对齐的, 因此, 可支持的跳转范围实际上是 $2 \times 18 = 256$ KB(相对PC的-128KB~+128KB), 如果确定跳转目标, 地址在分支指令前后的128KB范围内, 编译器就可以编码只生成一条简单的条件分支指令



扫码观看
视频讲解更清晰

J-type

OP(6)	target
-------	--------

跳转指令(j): 由线的32位指令长度对于大型程序的分支跳转支持确实是个难题，MIPS指令中最小的操作码域占6位，剩下的26位用于跳转目标的编址，由于所有指令在内存中都是4字节对齐的，因此最低的两个，比特位是无需存储的，这样实际可供寻址范围为 $2 \times 2^8 = 256$ MB分支跳转地址被当作一个256 MB的段内绝对地址，而非PC相对寻址，这对于地址范围超过256 MB的跳转程序而言，是无能为力的，所幸目前也很少遇到这么大的远程跳转需求

补充: 段外分支跳转可以使用寄存器跳转指令，实现它可以跳转到任意有效的32位地址

【题1】 MIPS指令分为R、 I、 J三种类型的指令，下列关于MIPS指令格式的描述中，正确的是(ABCD) (多选)

- A、 指令长度固定
- B、 操作码字段长度固定
- C、 指令中寄存器字段长度固定
- D、 立即数字段长度固定



扫码观看
视频讲解更清晰

考点2 MIPS源程序框架

#说明程序的目的和作用

.data #定义数据段

... #数据变量声明

.text #定义代码段

.globl main

main: #程序执行起始位置

... #主程序(指令代填)

li \$v0, 10 #程序正常执行终止

syscall #汇编结束

data:定义数据段

.test:定义代码段

.global:定义全局变量



扫码观看
视频讲解更清晰

考点3 系统调用Syscall

.data #数据段

str: .asciiz " Hello MIPS Assmbly! \n " #定义字符串str

.text #代码段

la \$a0, str #将str的地址赋给寄存器\$a0

li \$v0, 4 #将寄存器\$v0赋值为4

syscall #调用syscall,打印字符串str

li \$v0, 10 #将寄存器\$v0赋值为10

syscall #调用syscall,退出程序



扫码观看
视频讲解更清晰

服务	功能调用码	所需参数	返回值
打印一个整型	\$v0 = 1	将要打印的整型赋值给 \$a0	
打印一个浮点	\$v0 = 2	将要打印的浮点赋值给 \$f12	
打印双精度	\$v0 = 3	将要打印的双精度赋值给 \$f12	
打印字符串	\$v0 = 4	将要打印的字符串的地址赋值 给 \$a0	
读取一个整型	\$v0 = 5		将读取的整型赋值 给 \$v0
读取浮点	\$v0 = 6		将读取的浮点赋值 给 \$v0
读取双精度	\$v0 = 7		将读取的双精度赋 值给 \$v0

服务	功能调用码	所需参数	返回值
读取字符串	\$v0 = 8	将读取的字符串地址赋值给 \$a0 将读取的字符串长度赋值给 \$a1	
同C中的sbrk()函数动态分配内存	\$v0 = 9	\$a0 = amount 需要分配的空间大小（单位是字节 bytes）	address in \$v0 将分配好的空间首地址给 \$v0
退出	\$v0 = 10		