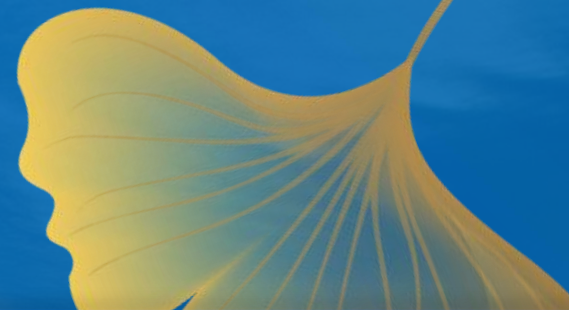




南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022

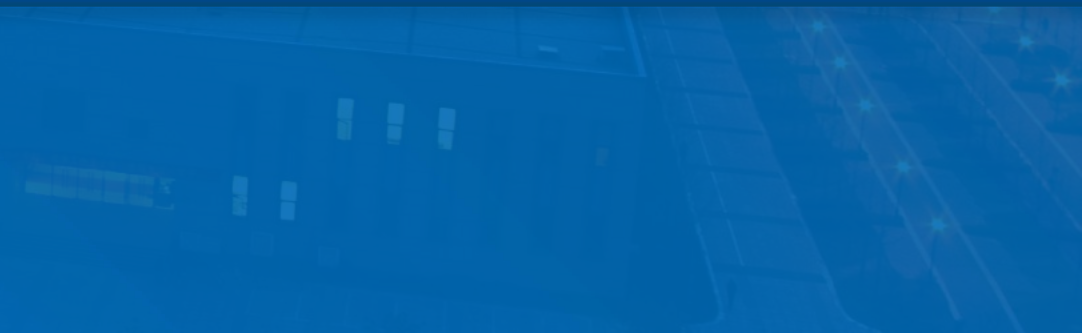
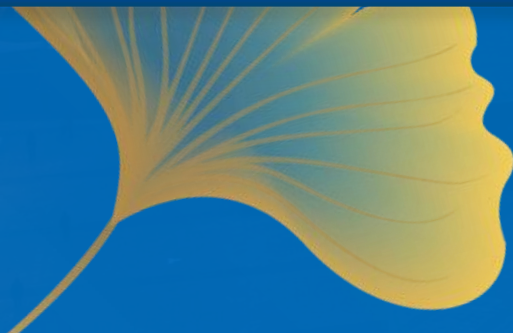
1902 2022



机器语言程序设计

Machine Language Programming

李杉杉





南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022

CONTENT

目录

01 指令集架构回顾

02 机器语言程序设计

03 示例





南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



指令集架构

指令集架构

- 计算机硬件和软件之间的接口
- 计算机能够执行的指令集合
 - 操作码：让计算机执行的操作
 - 操作数：每一步操作所需的数据
 - “数据类型”：操作数在计算机中的表示方式
 - “寻址模式”：如何计算操作数在存储器中的地址
 - 存储器
 - 地址空间：计算机存储单元的数量 (2^n)
 - 寻址能力：每个存储单元存储信息的能力 (m 位)
 - 寄存器集 (DLX包含32个整数寄存器)

DLX
算术/逻辑运算指令
数据传送指令
控制指令
浮点指令

DLX
二进制补码整数
(8/16/32位)
单/双精度浮点数
(32/64位)

DLX
基址+偏移量

不同的指令集结构规定的操作、操作数数据类型和寻址模式等是不同的。



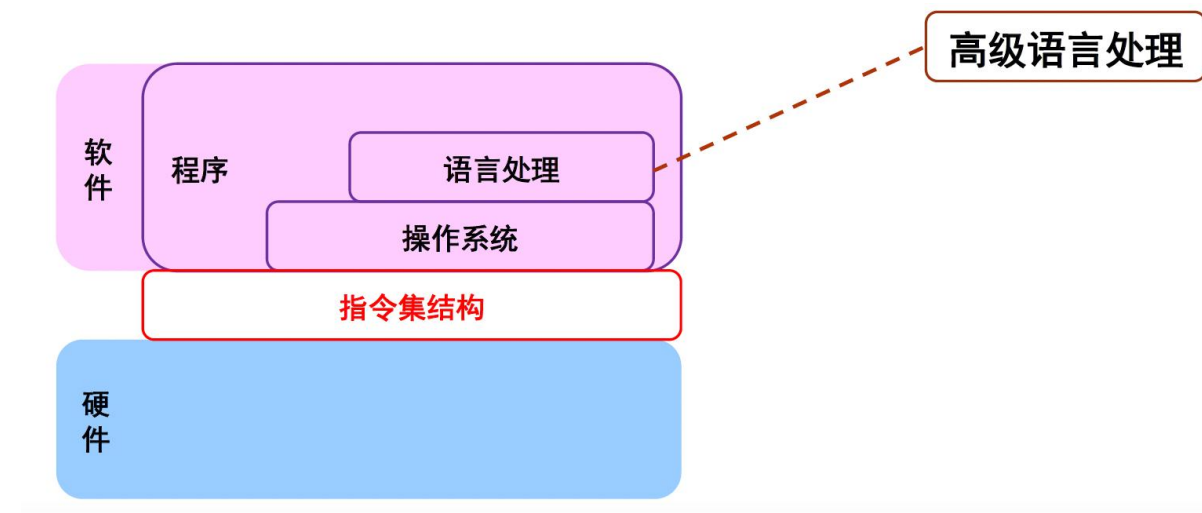
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



高级语言

高级语言

- 与底层计算机指令集无关
- “独立于机器”
- 不能直接被计算机执行
- 被翻译为目标机器ISA的二进制指令序列





南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



低级语言

低级语言

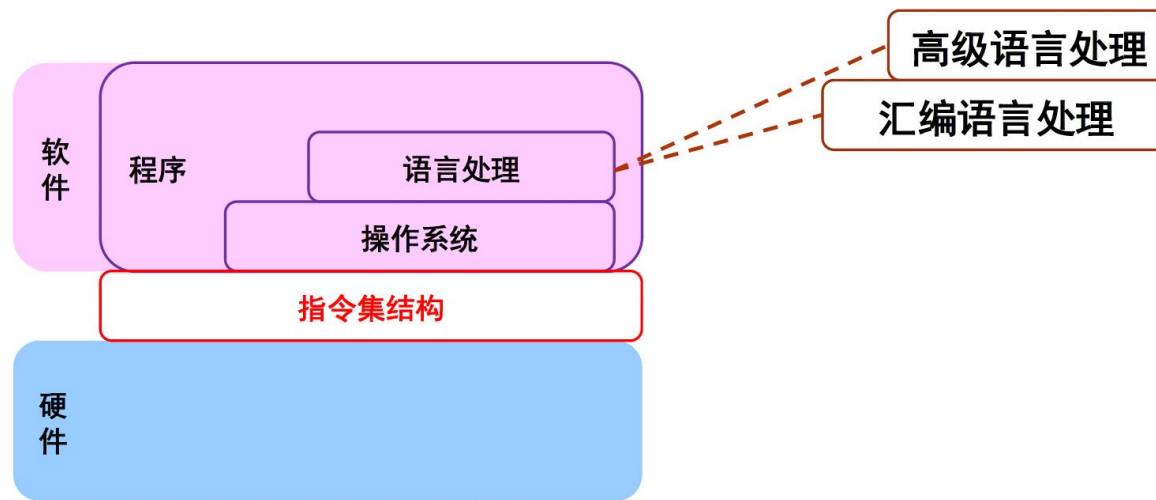
- 与执行程序的计算机指令集紧密相关

- **机器语言**

依据指令集使用二进制编码，直接在计算机上执行，不需要经过语言处理

- **汇编语言**

依据指令集的汇编语言格式编写，需经过语言处理，翻译为机器语言才能在计算机上执行





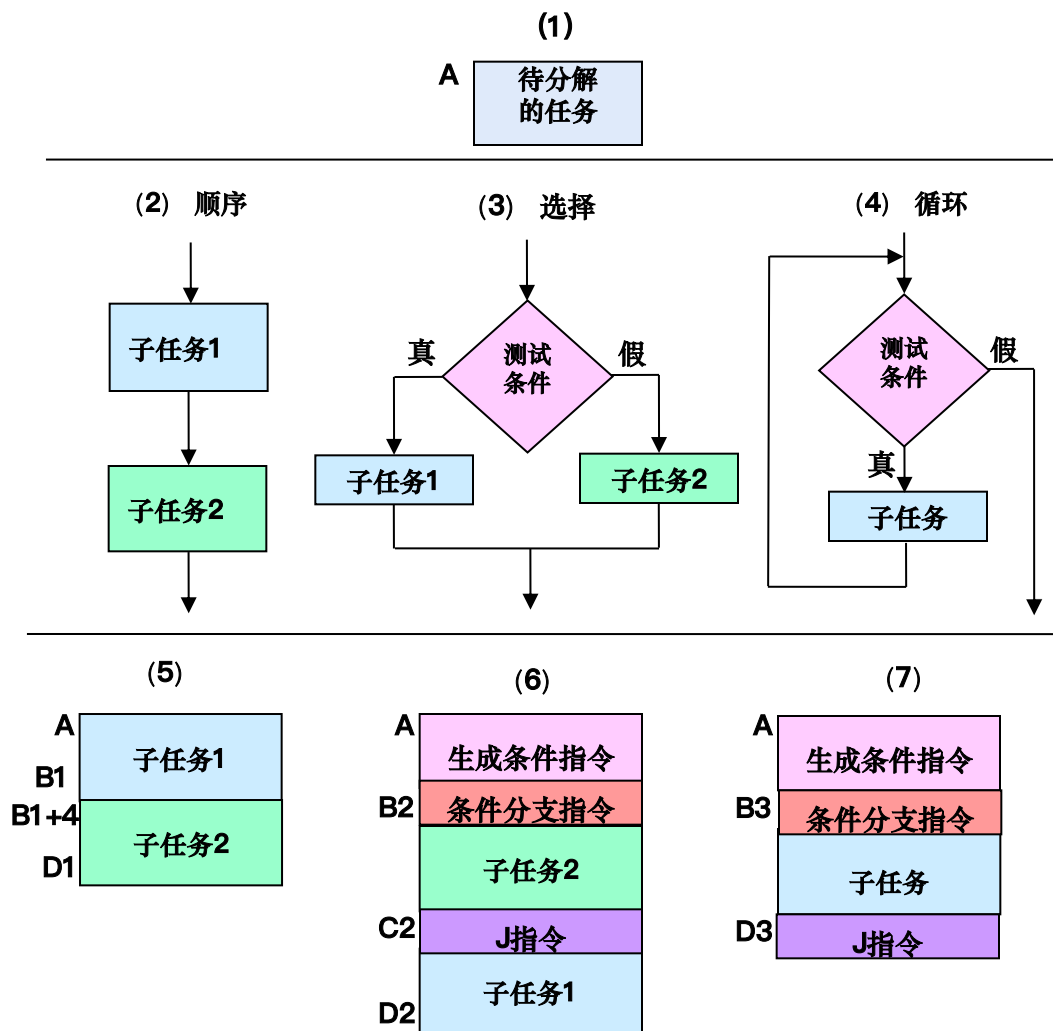
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



结构化程序设计

三种基本结构

- 顺序
- 选择
- 循环





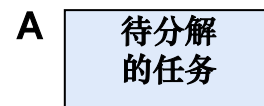
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



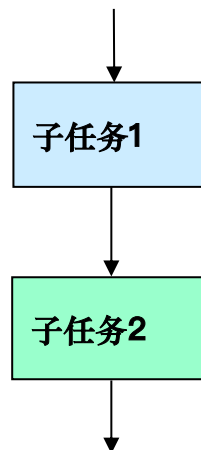
结构化程序设计

顺序结构

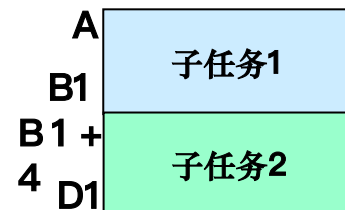
(1)



(2) 顺序



(5)





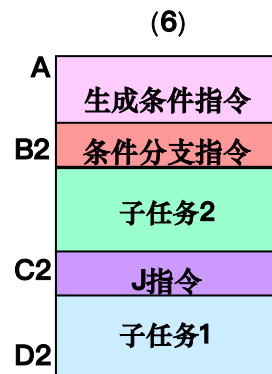
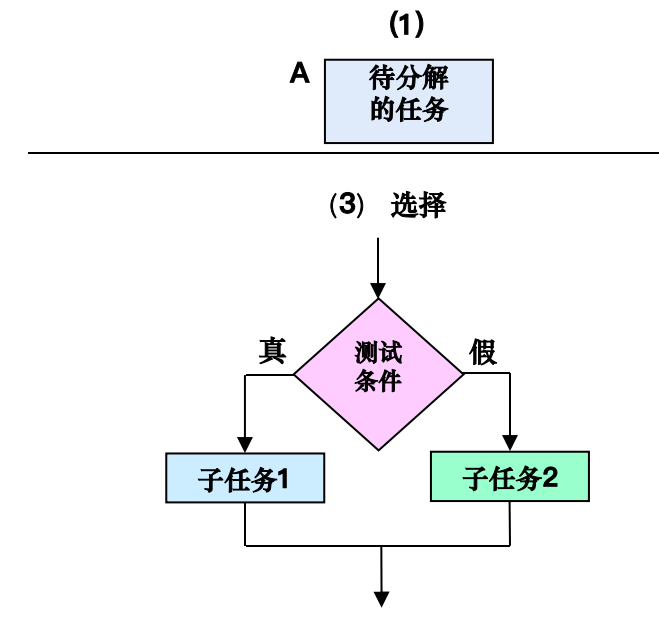
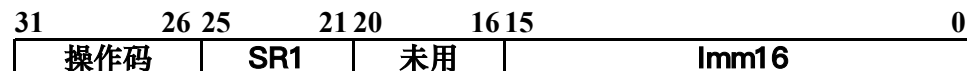
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



结构化程序设计

选择结构

- 一组指令序列生成条件
 - 将某个寄存器Rx设置为零（假）/非零（真）
- 地址B2“条件分支指令”测试该寄存器
 - 条件为真（BNEZ Rx, Y）
 - $PC \leftarrow C2 + 4$
 - 立即数Y:子任务2的指令数目加1后再乘以4
 - 条件为假
 - $PC \leftarrow B2 + 4$
 - 子任务2
 - 终止于C2中的无条件跳转指令
 - $PC \leftarrow D2 + 4$
 - J指令中的立即数: 子任务1的指令数目乘以4





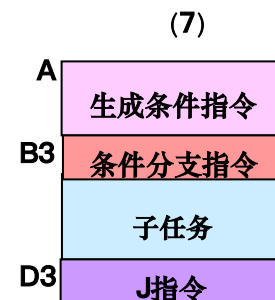
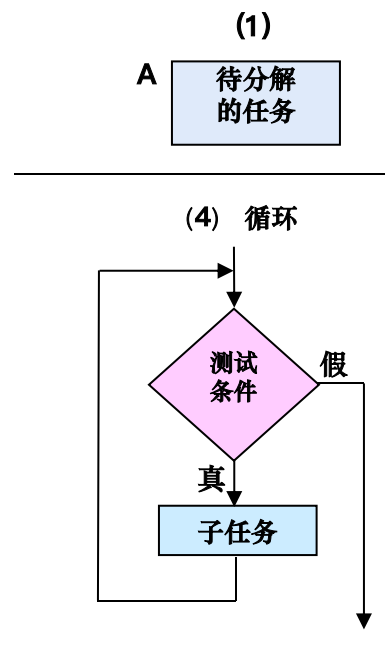
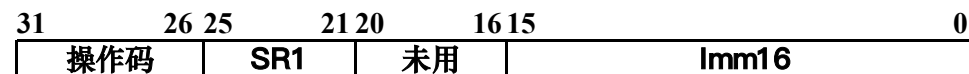
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



结构化程序设计

循环结构

- 一组指令序列生成条件
 - 将某个寄存器Rx设置为零（假）/非零（真）
- 地址B3“条件分支指令”测试该寄存器
 - 条件为假（BEQZ Rx, Y）
 - $PC \leftarrow D3+4$
 - 立即数Y：子任务的指令数目加1后再乘以4
 - 条件为真
 - $PC \leftarrow B3+4$
 - 子任务
 - 结束于D3中的无条件跳转指令
 - $PC \leftarrow A$
 - 问题：J指令中的立即数应为多少？





南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



结构化程序设计

示例：文档加密

- 根据键盘输入的数值 n （0到9之间的整数，ASCII码x30~x39），对文档进行加密
- 文档有一个个字符组成，终止标识EOT(x04)
- 加密算法：
 - 如果文档中的字符ASCII码值大于“ $126-n$ ”，将该字符减去“ $94-n$ ”，并替换原来的字符；
 - 而其他字符则加上 n ，进行替换；
 - 最后在显示器上显示字符“Y” (x59)，加密结束。
 - 假设文档中的字符ASCII码值在33~126范围内。



南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：文档加密

字符	ASCII		字符	ASCII		字符	ASCII		字符	ASCII	
	D	H		D	H		D	H		D	H
NUL	0	00	SP	32	20	@	64	40	`	96	60
SOH	1	01	!	33	21	A	65	41	a	97	61
STX	2	02	"	34	22	B	66	42	b	98	62
ETX	3	03	#	35	23	C	67	43	c	99	63
EOT	4	04	\$	36	24	D	68	44	d	100	64
ENQ	5	05	%	37	25	E	69	45	e	101	65
ACK	6	06	&	38	26	F	70	46	f	102	66
BEL	7	07	'	39	27	G	71	47	g	103	67
BS	8	08	(40	28	H	72	48	h	104	68
HT	9	09)	41	29	I	73	49	i	105	69
LF	10	0A	*	42	2A	J	74	4A	j	106	6A
VT	11	0B	+	43	2B	K	75	4B	k	107	6B
FF	12	0C	,	44	2C	L	76	4C	l	108	6C
CR	13	0D	-	45	2D	M	77	4D	m	109	6D
SO	14	0E	.	46	2E	N	78	4E	n	110	6E
SI	15	0F	/	47	2F	O	79	4F	o	111	6F
DLE	16	10	0	48	30	P	80	50	p	112	70
DC1	17	11	1	49	31	Q	81	51	q	113	71
DC2	18	12	2	50	32	R	82	52	r	114	72
DC3	19	13	3	51	33	S	83	53	s	115	73
DC4	20	14	4	52	34	T	84	54	t	116	74
NAK	21	15	5	53	35	U	85	55	u	117	75
SYN	22	16	6	54	36	V	86	56	v	118	76
ETB	23	17	7	55	37	W	87	57	w	119	77
CAN	24	18	8	56	38	X	88	58	x	120	78
EM	25	19	9	57	39	Y	89	59	y	121	79
SUB	26	1A	:	58	3A	Z	90	5A	z	122	7A
ESC	27	1B	;	59	3B	[91	5B	{	123	7B
FS	28	1C	<	60	3C	\	92	5C		124	7C
GS	29	1D	=	61	3D]	93	5D	}	125	7D
RS	30	1E	>	62	3E	^	94	5E	~	126	7E
US	31	1F	?	63	3F	_	95	5F	DEL	127	7F



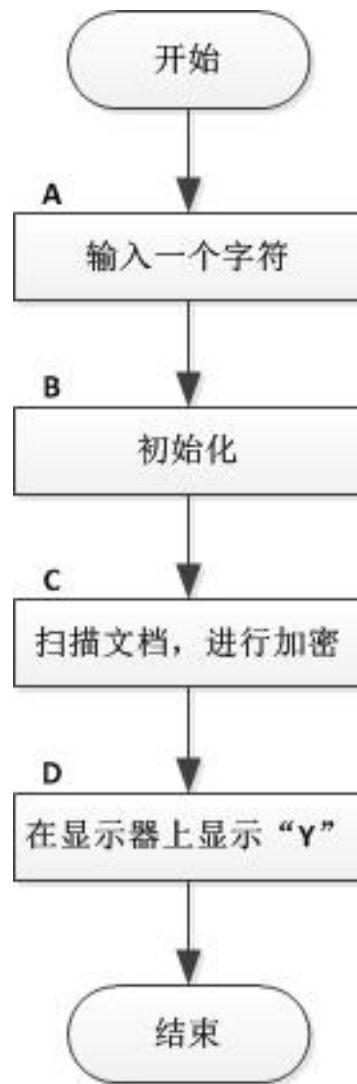
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：文档加密

系统分解过程

- 分解为由4个子任务组成的顺序结构
 - 初始化：得到数值 n ，将指针指向被检查文档中第一个字符的地址，然后从被检查文档中提取第一个字符。





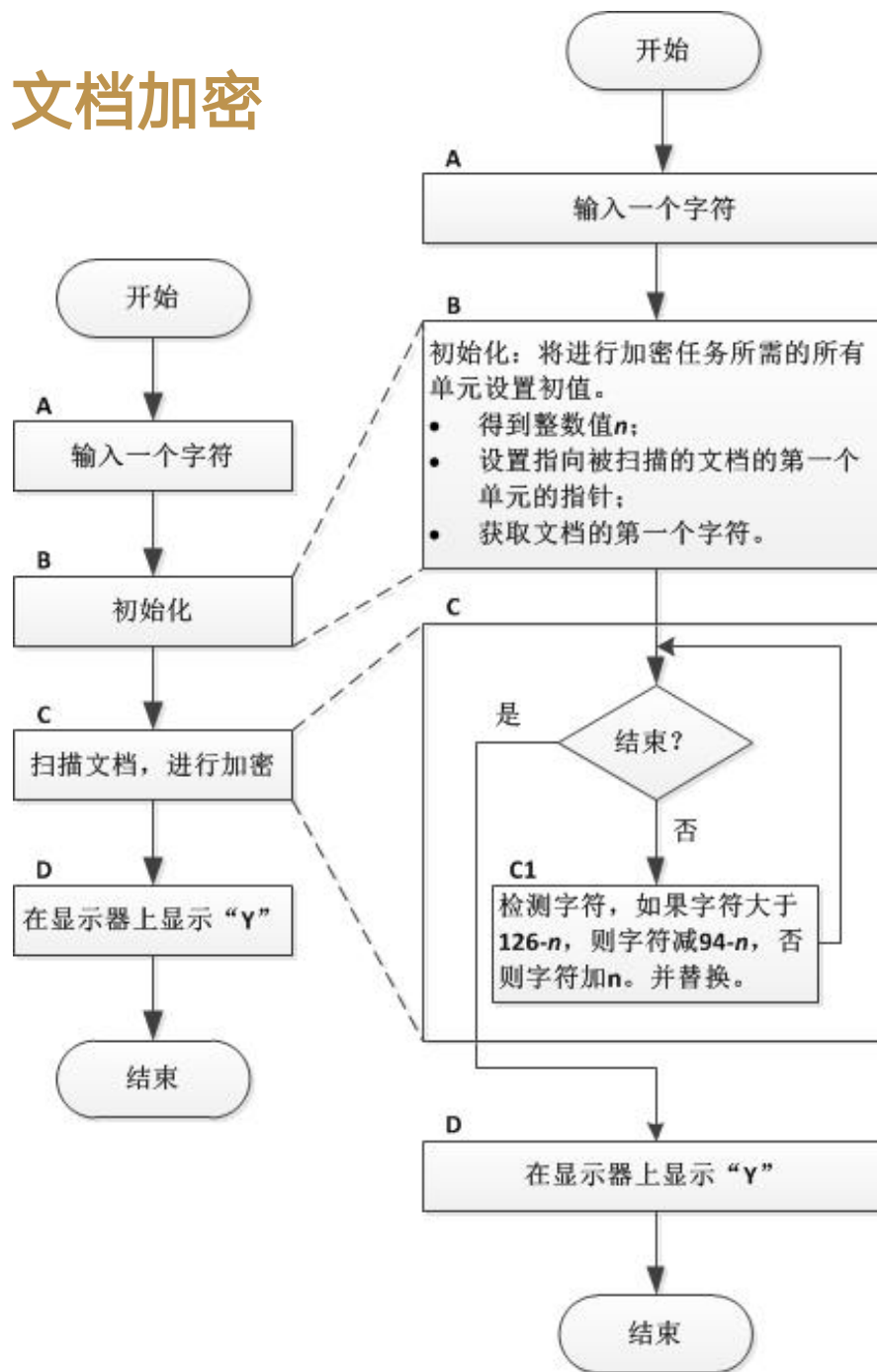
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：文档加密

分解C

- 循环结构：只要该文档还有字符需要加密
 - 文档结束，标志为EOT（传输结束，ASCII码为00000100）





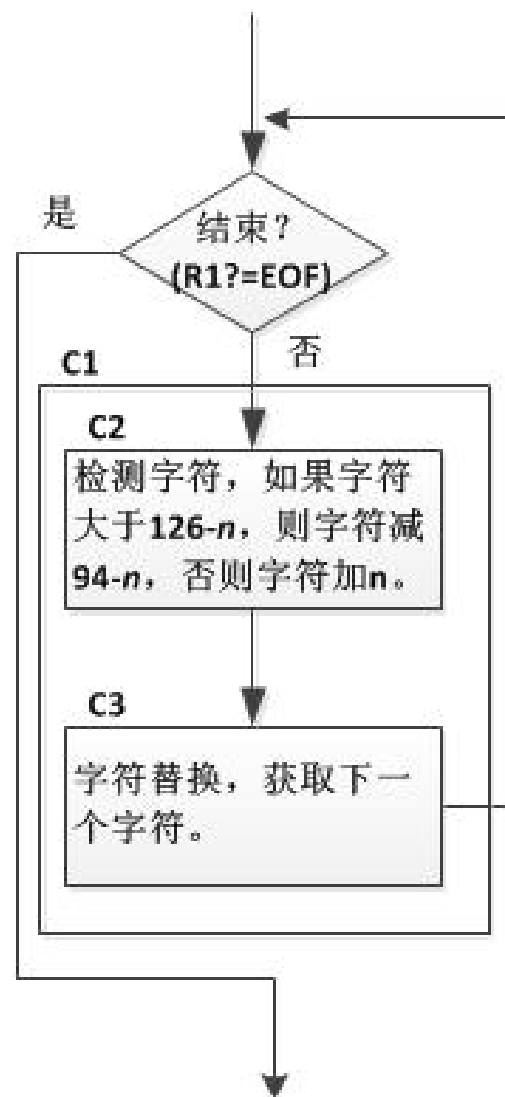
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：文档加密

分解C1

- 两个顺序的子任务C2和C3





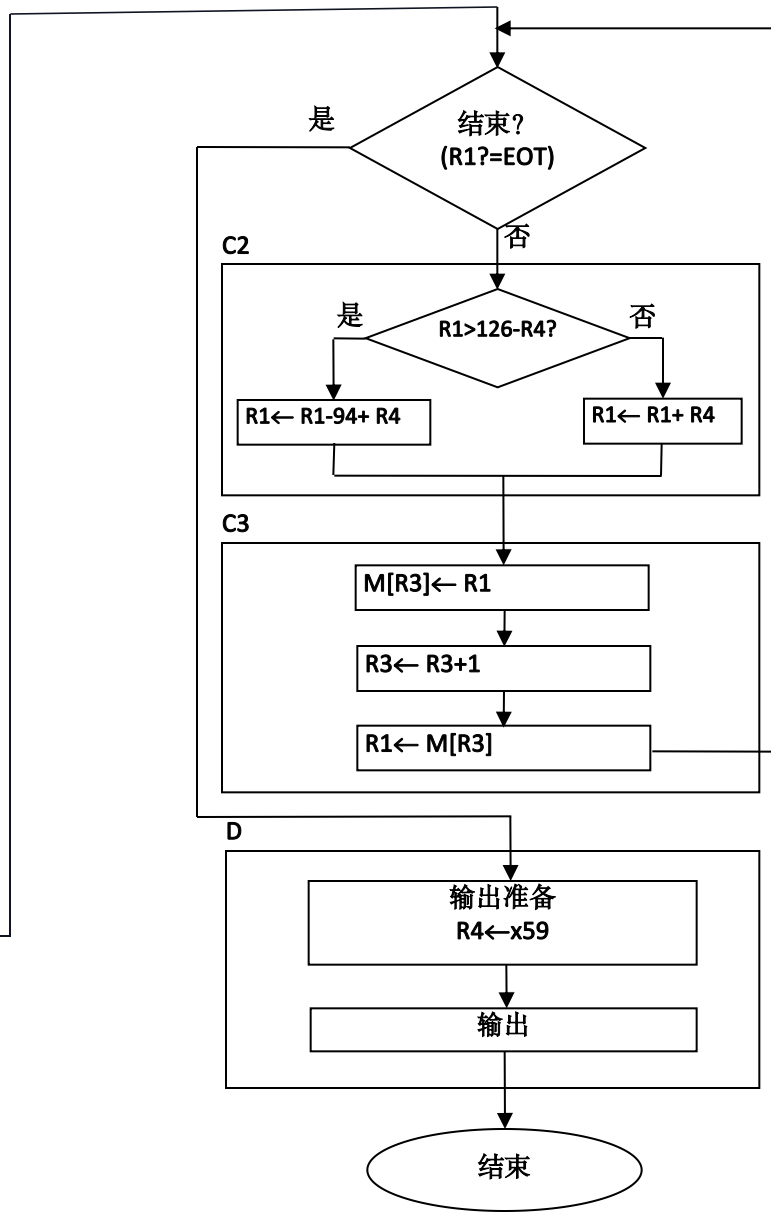
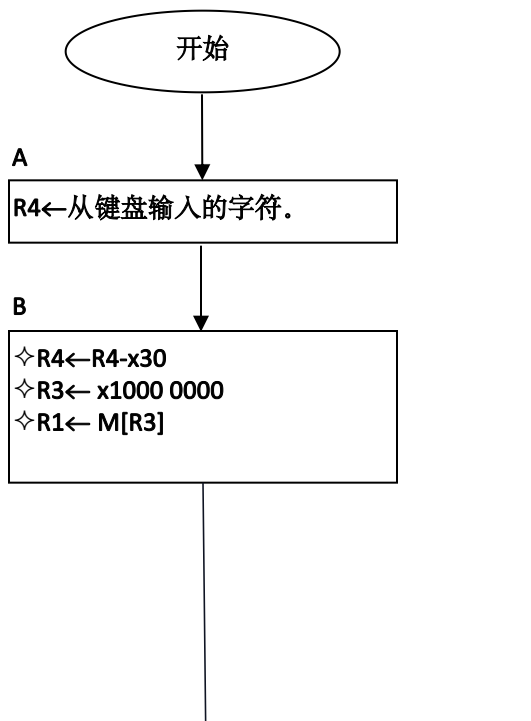
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：文档加密

分解C1

- 使用选择结构代替C2





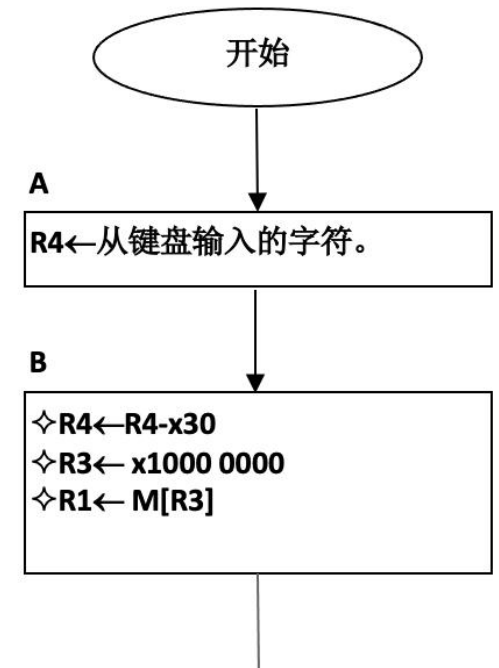
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：文档加密

顺序结构

地址	31	26	25	21	20	16	15	11	10	6	5	0	
x0400 0000	110000	000000 0000 0000 0000 0110											TRAP x06/IN
x0400 0004	000011	00100	00100	0000 0000 0011 0000									SUBI R4, R4, x30
x0400 0008	001100	00000	00011	0001 0000 0000 0000									LHI R3, x1000
x0400 000C	010110	00011	00001	0000 0000 0000 0000									LB R1, 0(R3)
x0400 0010	010100	00001	00010	0000 0000 0000 0100									SEI R2, R1, #4
x0400 0014	101001	00010	00000 0000 0011 0000									BNEZ R2, x30	
x0400 0018	000001	00000	00101	0000 0000 0111 1111									ADDI R5, R0, x7F
x0400 001C	000000	00101	00100	00101	000000			000011			SUB R5, R5, R4		
x0400 0020	000000	00001	00101	00010	000000			010000			SLT R2, R1, R5		
x0400 0024	101001	00010	00000 0000 0000 1100									BNEZ R2, x0C	
x0400 0028	000011	00101	00101	0000 0000 0010 0001									SUBI R5, R5, x21
x0400 002C	000000	00001	00101	00001	000000			000011			SUB R1, R1, R5		
x0400 0030	101100	000000 0000 0000 0000 0100											J x04
x0400 0034	000000	00001	00100	00001	000000			000001			ADD R1, R1, R4		
x0400 0038	010111	00011	00001	0000 0000 0000 0000									SB 0(R3), R1
x0400 003C	000001	00011	00011	0000 0000 0000 0001									ADDI R3, R3, #1
x0400 0040	010110	00011	00001	0000 0000 0000 0000									LB R1, 0(R3)
x0400 0044	101100	111111 1111 1111 1111 1100 1000											J #-56
x0400 0048	000001	00000	00100	0000 0000 0101 1001									ADDI R4, R0, x59
x0400 004C	110000	000000 0000 0000 0000 0000 0111											TRAP x07/OUT
x0400 0050	110000	000000 0000 0000 0000 0000 0000											TRAP x00/HALT



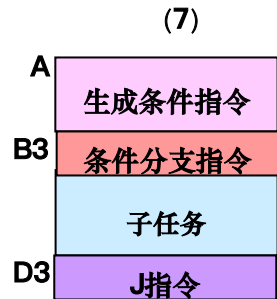
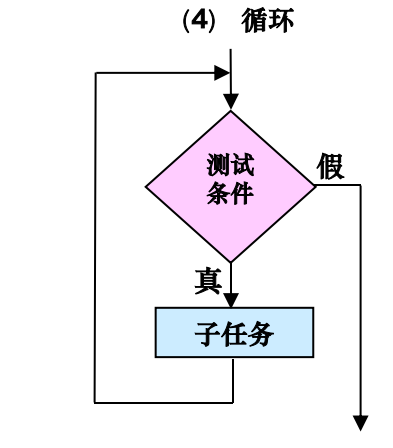
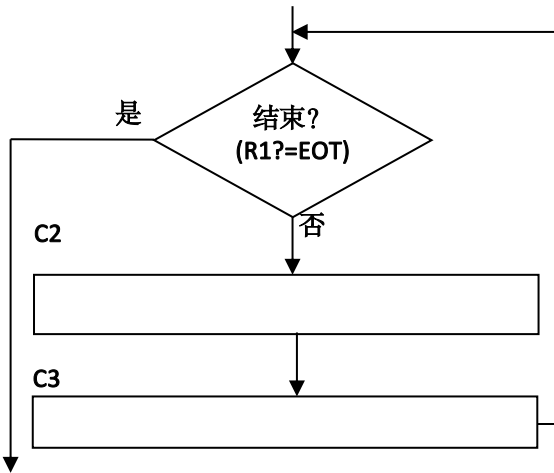


南京大學120周年校慶
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：文档加密

循环结构

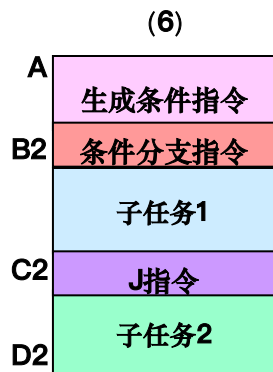
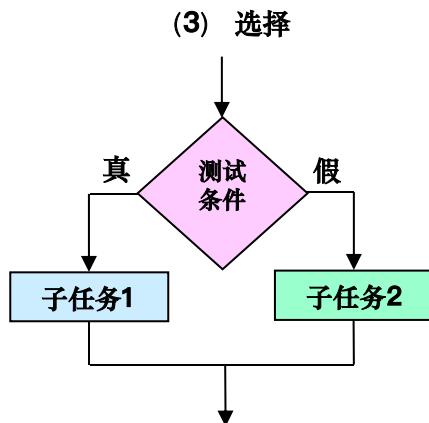
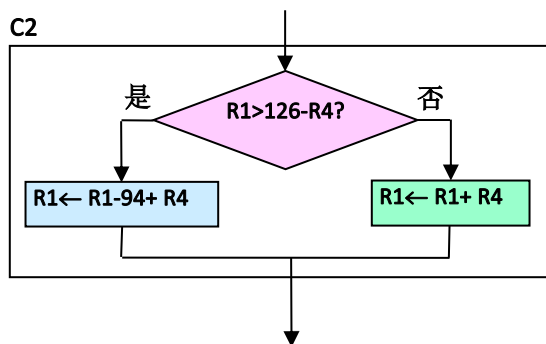


地址	31	26	25	21	20	16	15	11	10	6	5	0	
x0400 0000	110000		000000 0000 0000 0000 0000 0110								TRAP x06/IN		
x0400 0004	000011		00100		00100		0000 0000 0011 0000						SUBI R4, R4, x30
x0400 0008	001100		00000		00011		0001 0000 0000 0000						LHI R3, x1000
x0400 000C	010110		00011		00001		0000 0000 0000 0000						LB R1, 0(R3)
x0400 0010	010100		00001		00010		0000 0000 0000 0100						SEQI R2, R1, x04
x0400 0014	101001		00010		00000 0000 0011 0000						BNEZ R2, x30		
x0400 0018	000001		00000		00101		0000 0000 0111 1111						ADDI R5, R0, x7F
x0400 001C	000000		00101		00100		00101		000000		000011		SUB R5, R5, R4
x0400 0020	000000		00001		00101		00010		000000		010000		SLT R2, R1, R5
x0400 0024	101001		00010		00000 0000 0000 1100								BNEZ R2, x0C
x0400 0028	000011		00101		00101		0000 0000 0010 0001						SUBI R5, R5, x21
x0400 002C	000000		00001		00101		00001		000000		000011		SUB R1, R1, R5
x0400 0030	101100		000000 0000 0000 0000 0000 0100										J x04
x0400 0034	000000		00001		00100		00001		000000		000001		ADD R1, R1, R4
x0400 0038	010111		00011		00001		0000 0000 0000 0000						SB 0(R3), R1
x0400 003C	000001		00011		00011		0000 0000 0000 0001						ADDI R3, R3, #1
x0400 0040	010110		00011		00001		0000 0000 0000 0000						LB R1, 0(R3)
x0400 0044	101100		111111 1111 1111 1111 1100 1000										J #-56
x0400 0048	000001		00000		00100		0000 0000 0101 1001						ADDI R4, R0, x59
x0400 004C	110000		000000 0000 0000 0000 0000 0111										TRAP x07/OUT
x0400 0050	110000		000000 0000 0000 0000 0000 0000										TRAP x00/HALT



示例：文档加密

C2选择结构



地址	31	26	25	21	20	16	15	11	10	6	5	0	
x0400 0000	110000	000000 0000 0000 0000 0000 0110										TRAP x06/IN	
x0400 0004	000011	00100	00100	0000 0000 0011 0000									SUBI R4, R4, x30
x0400 0008	001100	00000	00011	0001 0000 0000 0000									LHI R3, x1000
x0400 000C	010110	00011	00001	0000 0000 0000 0000									LB R1, 0(R3)
x0400 0010	010100	00001	00010	0000 0000 0000 0100									SEQI R2, R1, #4
x0400 0014	101001	00010	00000 0000 0011 0000										BEQZ R2, x30
x0400 0018	000001	00000	00101	0000 0000 0111 1111									ADDI R5, R0, x7F
x0400 001C	000000	00101	00100	00101	000000		000011						SUB R5, R5, R4
x0400 0020	000000	00001	00101	00010	000000		010000						SLT R2, R1, R5
x0400 0024	101001	00010	00000 0000 0000 1100										BNEZ R2, x0C
x0400 0028	000011	00101	00101	0000 0000 0010 0001									SUBI R5, R5, x21
x0400 002C	000000	00001	00101	00001	000000		000011						SUB R1, R1, R5
x0400 0030	101100	000000 0000 0000 0000 0000 0100											J x04
x0400 0034	000000	00001	00100	00001	000000		000001						ADD R1, R1, R4
x0400 0038	010111	00011	00001	0000 0000 0000 0000									SB 0(R3), R1
x0400 003C	000001	00011	00011	0000 0000 0000 0001									ADDI R3, R3, #1
x0400 0040	010110	00011	00001	0000 0000 0000 0000									LB R1, 0(R3)
x0400 0044	101100	111111 1111 1111 1111 1100 1000											J #-56
x0400 0048	000001	00000	00100	0000 0000 0101 1001									ADDI R4, R0, x59
x0400 004C	110000	000000 0000 0000 0000 0000 0111											TRAP x07/OUT
x0400 0050	110000	000000 0000 0000 0000 0000 0000											TRAP x00/HALT



南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：判断连续存储单元内是否包含5

检查

- 从地址x3000 0000开始存储的10个整数
 - 有5，R1设置为1
 - 没有5，R1为0

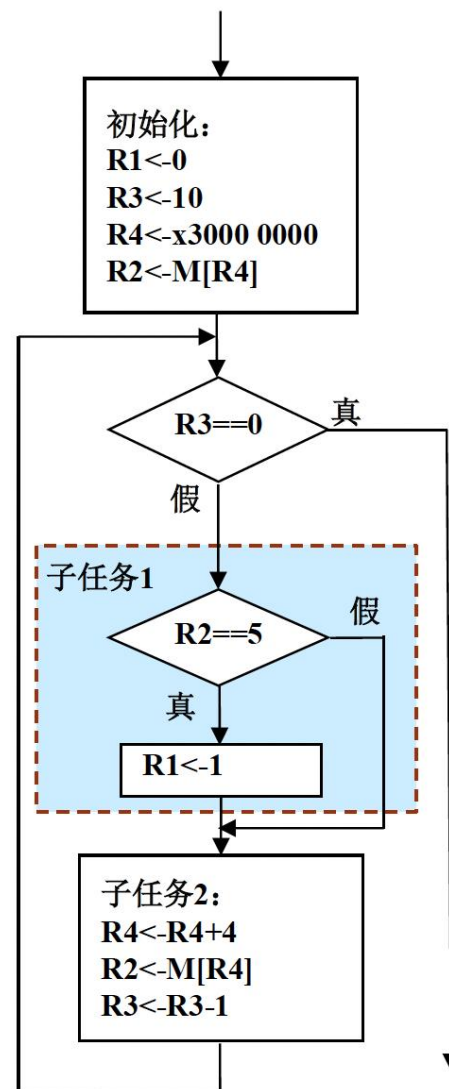


南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：判断连续存储单元内是否包含5

- 计数器控制的循环
 - R3, 计数器
- 子任务1
 - 选择结构





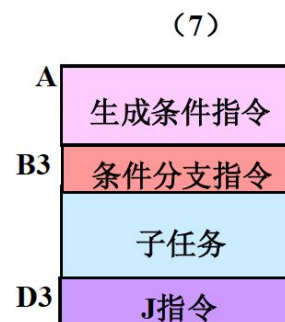
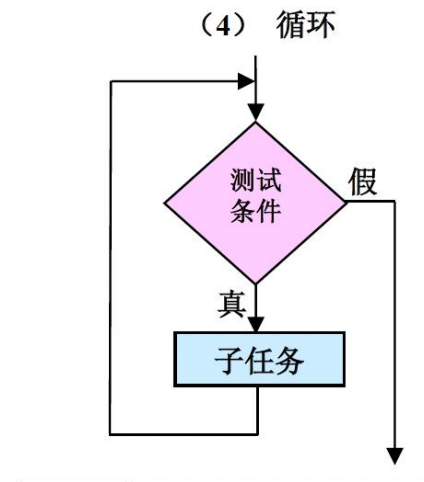
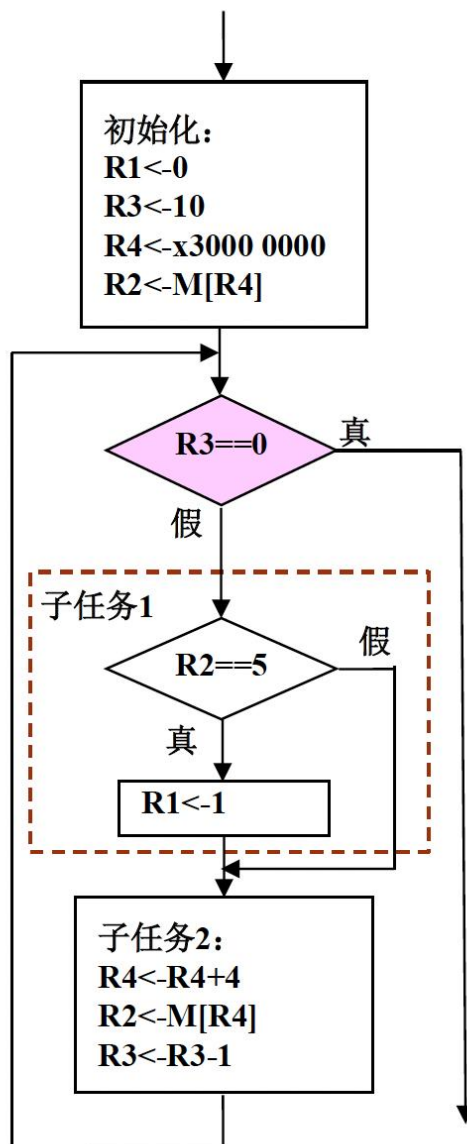
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：判断连续存储单元内是否包含5

测试条件 $R3==0$

- 不需要生成条件指令
- 条件分支指令
 - BEQZ R3,D3+4





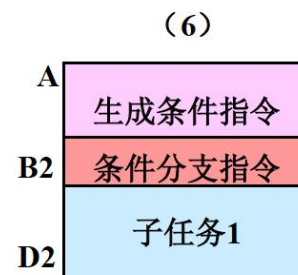
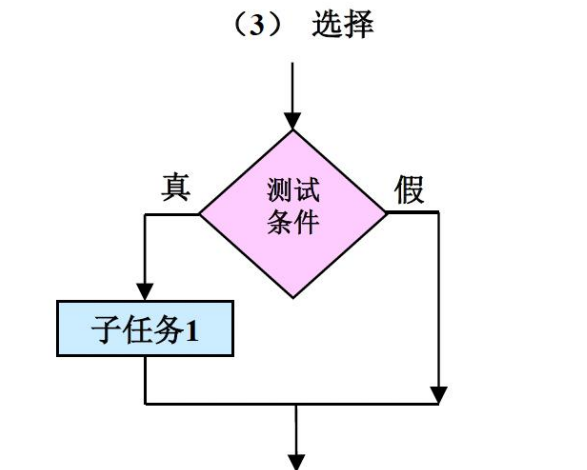
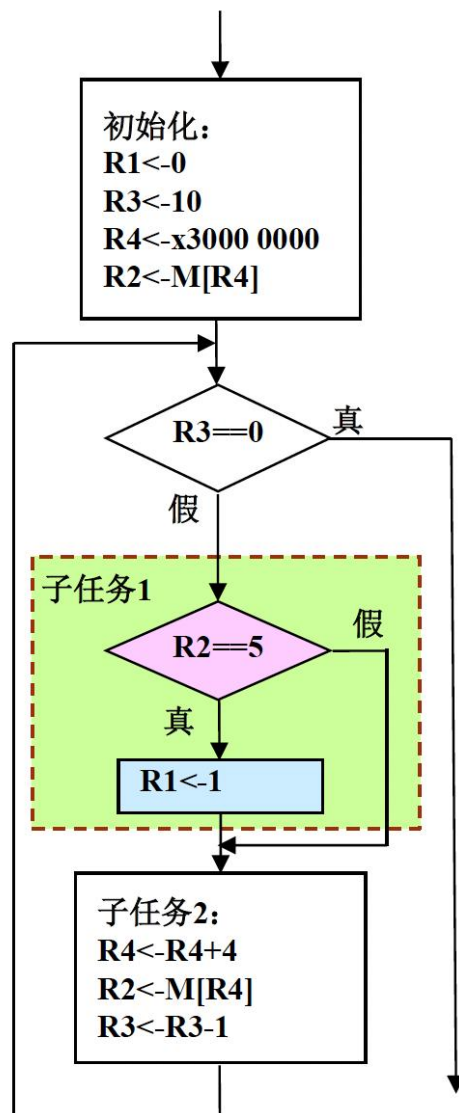
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：判断连续存储单元内是否包含5

测试条件 $R2 == 5$

- 生成条件指令
 - $\text{SEQI } R_x, R2, \#5$
- 条件分支指令
 - $\text{BEQZ } R_x, D2+4$





南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022

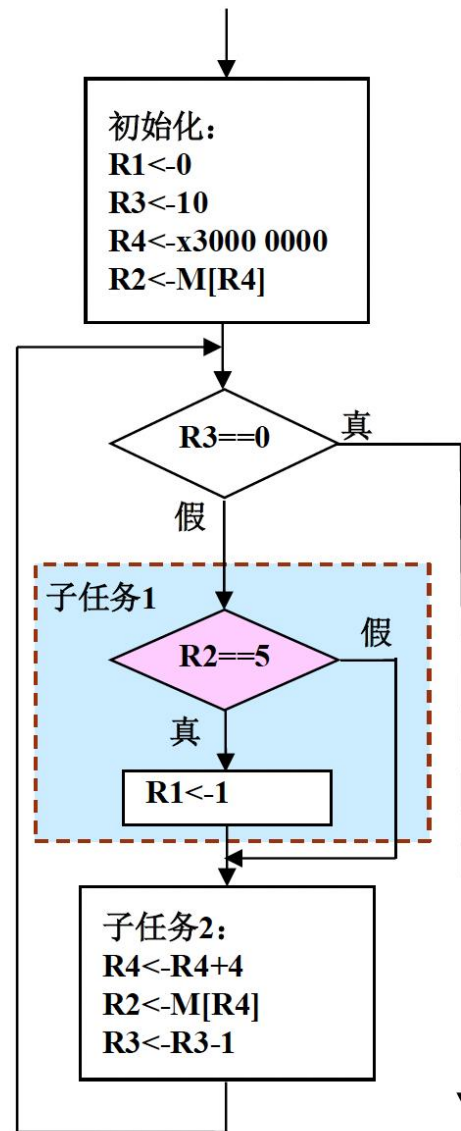


示例：判断连续存储单元内是否包含5

选择结构

- 当R2为5时，设置R1为1
- 使用J指令跳出循环
- 机器语言程序如下：

31	26	25	21	20	16	15	11	10	6	5	0	解释
001001		00001		00001		0000 0000 0000 0000						ANDI R1,R1, #0
000001		00000		00011		0000 0000 0000 1010						ADDI R3,R0, #10
001100		00000		00100		0011 0000 0000 0000						LHI R4, x3000
011100		00100		00010		0000 0000 0000 0000						LW R2, 0(R4)
101000		00011		00000		0000 0000 0010 0000						BEQZ R3, #32
010100		00010		00101		0000 0000 0000 0101						SEQI R5, R2, #5
101000		00101		00000		0000 0000 0000 1000						BEQZ R5, #8
000001		00000		00001		0000 0000 0000 0001						ADDI R1,R0, #1
101100		00 0000 0000 0000 0000 0001 0000										J #16
000001		00100		00100		0000 0000 0000 0100						ADDI R4,R4, #4
011100		00100		00010		0000 0000 0000 0000						LW R2, 0(R4)
000011		00011		00011		0000 0000 0000 0001						SUBI R3,R3, #1
101100		11 1111 1111 1111 1111 1101 1100										J #-36
.....												





南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：找到字中的第一个“1”

检查

- x3000 0000~x3000 0003中的字
- 找出第一个“1”（从左到右）
 - 存储到R1中
- 如果没有1
 - $R1 \leftarrow -1$
- 例如
 - 0010 0000 0000 0000 0000 0000 0000 0000, $R1=29$
 - 0000 0000 0000 0000 0000 0000 0010 0000, $R1=5$



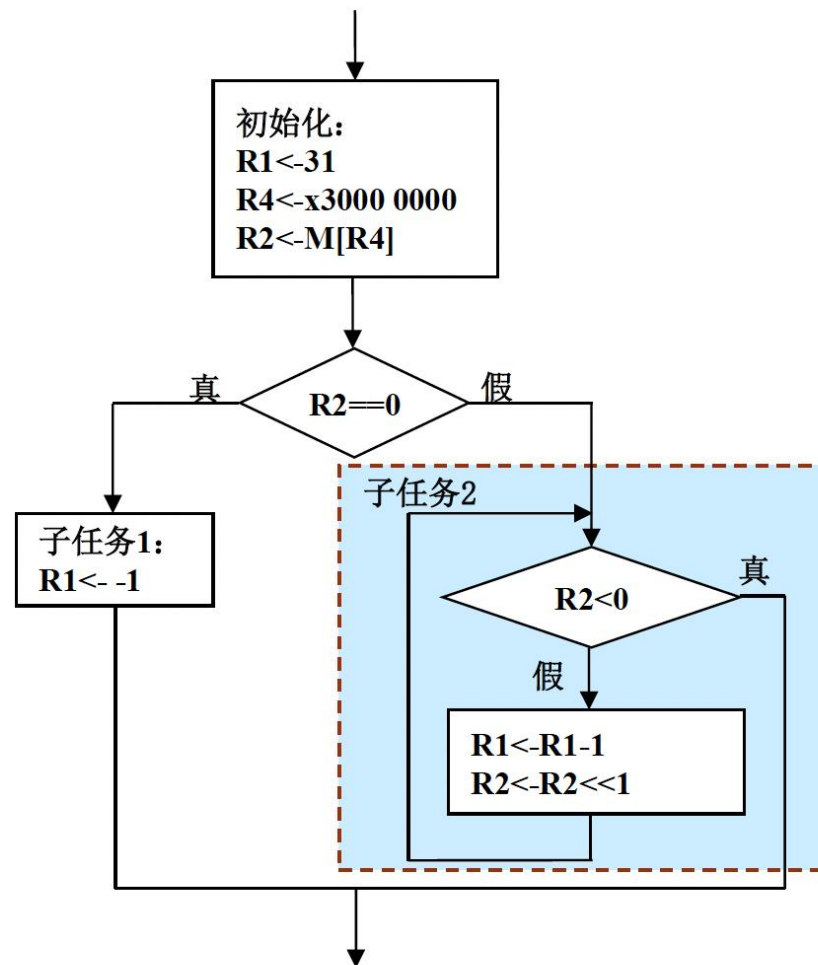
南京大学120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：找到字中的第一个“1”

选择结构

- 子任务2
 - 标志控制的循环
 - 标志
 - $R2 < 0$: $R2[31] = 1$
 - 循环子任务
 - $R2 = R2 \ll 1$
 - $R2[30], R2[29] \dots == 1$?





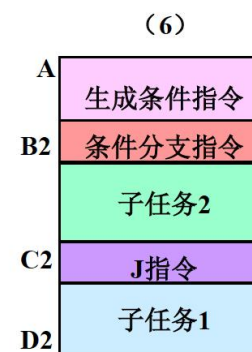
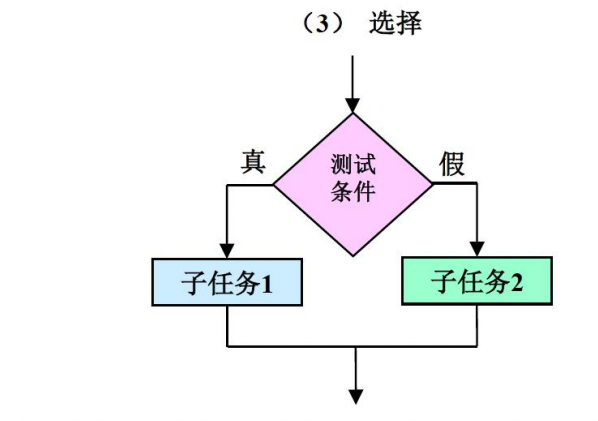
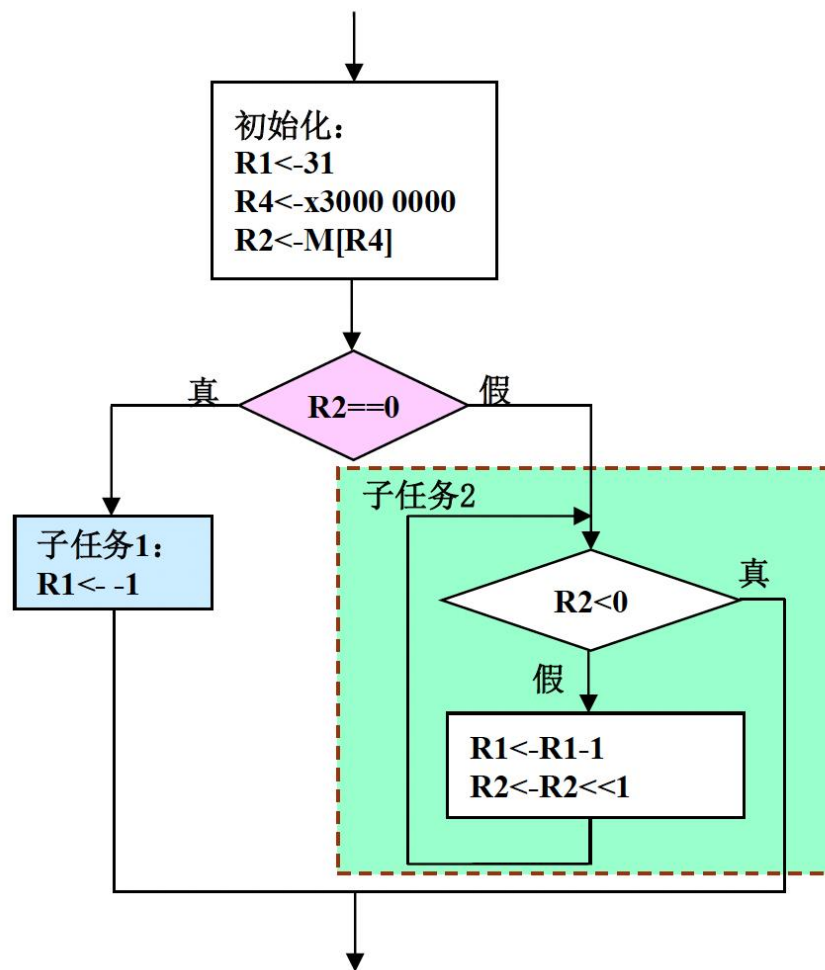
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：找到字中的第一个“1”

测试条件 $R2 == 0$

- 不需要生成条件指令
- 条件分支指令
 - BEQZ R2,C2+4





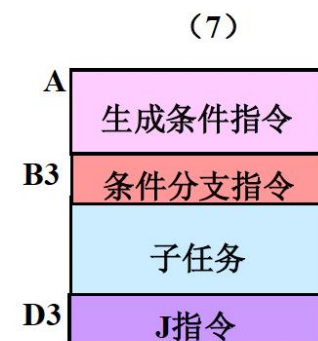
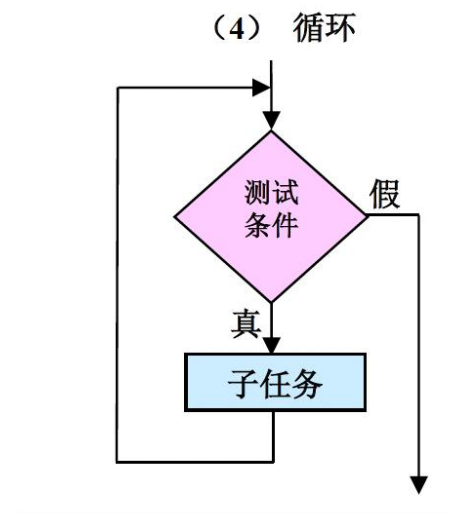
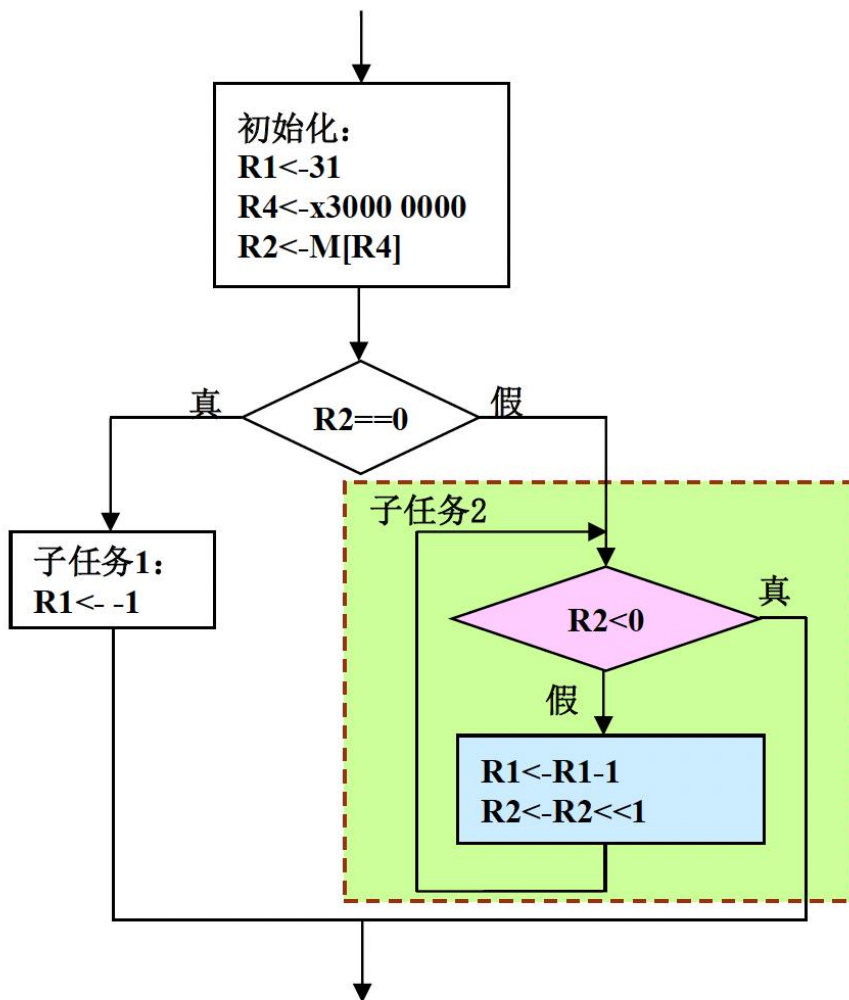
南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



示例：找到字中的第一个“1”

测试条件 $R2 < 0$

- 生成条件指令
 - $\text{SLTI } R_x, R2, \#0$
- 条件分支指令
 - $\text{BNEZ } R_x, D3+4$





机器语言程序

31	26	25	21	20	16	15	11	10	6	5	0	解释
000001		00000		00001		0000 0000 0001 1111						ADDI R1,R0, #31
001100		00000		00100		0011 0000 0000 0000						LHI R4, x3000
011100		00100		00010		0000 0000 0000 0000						LW R2, 0(R4)
101000		00010		00000		0000 0000 0001 0100						BEQZ R2,#20
010000		00010		00011		0000 0000 0000 0000						SLTI R3, R2, #0
101001		00011		00000		0000 0000 0001 0000						BNEZ R3, #16
000011		00001		00001		0000 0000 0000 0001						SUBI R1,R1, #1
001101		00010		00010		0000 0000 0000 0001						SLLI R2,R2, #1
101100		111111 1111 1111 1111 1110 1100										J #-20
001010		00000		00001		1111 1111 1111 1111						ORI R1,R0, #-1
.....												



南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



测试和调试

- 在第十一章 DLX汇编语言编程后，再做介绍



南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



练习

- 如下表存储器所示，当一段起始于单元x30000000的程序执行结束后，R1 ~ R6 的值分别是多少？

地址	数据
x30000000	00110000000000010100000000000000
x30000004	01110000001000100000000000000100
x30000008	01011000001000110000000000000100
x3000000C	01110100001000100000000000000000
x30000010	01110000010001000000000000000100
x30000014	01011000001001010000000000000001
x30000018	01011100010001000000000000000011
x3000001C	01110000001001100000000000000000
x30000020	01110000110001100000000000000000
x30000024	11000000000000000000000000000000
.....
x40000000	10000111011001010100001100100001
x40000004	01000011001000010000000000000000
.....
x43210000	00000000000000000100000000000000
x43210004	00000000000000000100000000000000



南京大學120周年校庆
120th ANNIVERSARY
NANJING UNIVERSITY
1902 - 2022



习题

- 书面作业
 - 9.9
 - 9.10
 - 9.11
 - 10.1
 - 10.2



谢 谢

诚耀百廿 雄创一流

