

# Contents

<b>Esercitazione</b>	<b>1</b>
23/09/20 . . . . .	1
1 . . . . .	1
Soluzione . . . . .	1
25/09/20 . . . . .	3
1 . . . . .	3
Soluzione . . . . .	3

## Esercitazione

23/09/20

1

Dati i seguenti moduli:

- Modulo “main”:

```
.data
STRUCT: .space 20
VECT: .space 12
INT: .int 23

.text
.globl MAIN

MAIN:
    li $t0, 0xFFFF0ABCC
    sw $t0, STRUCT
    lw $t1, VECT
    beq $t0, $t0, MODULE
MAINEND:
    syscall
```

- Modulo “module”:

```
.data
ALPHA: .byte 'Y'

.text
.globl MODULE
RESTART:
    lw $t3, INT
MODULE:
    lb $t4, ALPHA
    sub $t4, $t4, $t3
    beq $t4, $0, RESTART
MODEND:
    j MAINEND
```

Si compilino le quattro tabelle relative a:

1. i moduli oggetto prodotti dall’assemblatore
2. le basi di rilocazione del codice e dei dati dei moduli
3. La tabella globale dei simboli
4. il contenuto del file eseguibile prodotto dal linker

## Soluzione

1. Tabella file oggetto

```

dim text: 18          | dim text: 14
dim data: 24          | dim data: 1

text:                 | text:
0  lui $t0, FFF0      | 0  lw $t3, 0000($gp)
4  ori $t0, $t0, ABCC | 4  lb $t4, 0000($gp)
8  sw $t0, 0000($gp)  | 8  sub $t4, $t4, $t3
C  lw $t1, 0000($gp)  | C  beq $t4, $0, FFFC
10 beq $t0, $t0, 0000 | 10 j 000 0000
14 syscall            |

data:                 | data:
0  uninitialized      | 0  0000 0059
14 uninitialized      |
20 0000 0017          |

symbols:              | symbols:
STRUCT D 0000 0000    | ALPHA D 0000 0000
VECT D 0000 0014      | RESTART T 0000 0000
INT D 0000 0020       | MODULE T 0000 0004
MAIN T 0000 0000      | MODEND T 0000 0010
MAINEND T 0000 0014   |

relocation:           | relocation:
8 sw STRUCT           | 0 lw INT
C lw VECT              | 4 lb ALPHA
10 beq MODULE          | 10 j MAINEND

```

## 2. Basi di rilocalizzazione

	main	module
base text	0040 0000	0040 0018
base data	1000 0000	1000 0024

## 3. Tabella globale dei simboli

simbolo	valore iniziale	base	valore finale
STRUCT	0000 0000	1000 0000	1000 0000
VECT	0000 0014	1000 0000	1000 0014
INT	0000 0020	1000 0000	1000 0020
MAIN	0000 0000	0040 0000	0040 0000
MAINEND	0000 0014	0040 0000	0040 0014
ALPHA	0000 0000	1000 0024	1000 0024
RESTART	0000 0000	0040 0018	0040 0018
MODULE	0000 0004	0040 0018	0040 001C
MODULEND	0000 0010	0040 0018	0040 0028

## 4. Eseguibile

```

0040 0000 lui $t0, FFF0
0040 0004 ori $t0, $t0, ABCC
0040 0008 sw $t0, 8000($gp)
0040 000C lw $t1, 8014($gp)
0040 0010 beq $t0, $t0, 0002
0040 0014 syscall
0040 0018 lw $t3, 8020($gp)
0040 001C lb $t4, 8024($gp)
0040 0020 sub $t4, $t4, $t3

```

```
0040 0024 beq $t4, $0, FFFC
0040 0028 j 010 0005
```

25/09/20

1

- Modulo “Main”:

```
.data
INT: .word 37
BLOCK: .space 12

.text
.globl MAIN
MAIN:
    addi $t0, $0, 0x100A
    sw $t0, INT
    la $t1, BLOCK
    lw $t2, ($t1)
    j LIBRARY
MAINEND:
    syscall
```

- Modulo “Library”:

```
.data
VAR: .space 4

.text
.globl LIBRARY
LIBRARY:
    lw $t3, VAR
    beq $t3, $t2, MAINEND
    addi $t3, $t3, 1
LIBEND:
    bne $t3, $t2, LIBRARY
    syscall
```

Si compilino le quattro tabelle relative a:

1. i moduli oggetto prodotti dall’assemblatore
2. le basi di rilocazione del codice e dei dati dei moduli
3. La tabella globale dei simboli
4. il contenuto del file eseguibile prodotto dal linker

## Soluzione

1. Tabella file oggetto

dim text: 1C		dim text: 14
dim data: 10		dim data: 4
text:		text:
0 addi \$t0, \$t0, 100A		0 lw \$t3, 0000(\$gp)
4 sw \$t0, 0000(\$gp)		4 beq \$t3, \$t2, 0000
8 lui \$t1, 0000		8 addi \$t3, \$t2, 0001
C ori \$t1, 0000		C bne \$t3, \$t2, FFFC
10 lw \$t2, 0000(\$gp)		10 syscall
14 j 000 0000		
18 syscall		
data:		data:
0 0000 0025		0 uninitialized

```

4  uninitialized      |
                           |
symbols:              | symbols:
INT      D  0000 0000 | VAR      D  0000 0000
BLOCK    D  0000 0004 | LIBRARY  T  0000 0000
MAIN     D  0000 0000 | LIBEND   T  0000 000C
MAINEND  T  0000 0018 |
                           |
relocation:          | relocation:
4  sw  INT           | 0   lw  VAR
8  lui $hi(BLOCK)    | 4   beq MAINEND
C  ori %lo(BLOCK)    |
14 j   LIBRARY       |

```

## 2. Basi di rilocalizzazione

	main	module
base text	0040 0000	0040 001C
base data	1000 0000	1000 0010

## 3. Tabella globale dei simboli

simbolo	valore iniziale	base	valore finale
INT	0000 0000	1000 0000	1000 0000
BLOCK	0000 0004	1000 0000	1000 0004
MAIN	0000 0000	0040 0000	0040 0000
MAINEND	0000 0018	0040 0000	0040 0018
VAR	0000 0000	1000 0010	1000 0010
LIBRARY	0000 0000	0040 001C	0040 001C
LIBEND	0000 000C	0040 001C	0040 0022

## 4. Eseguitibile

```

0040 0000: addi $t0, $0, 100A
0040 0004: sw $t0, 8000($gp)
0040 0008: lui $t1, 1000
0040 000C: ori $t1, $t1, 0004
0040 0010: lw $t2, 0000($t1)
0040 0014: j 010 0007
0040 0018: syscall
0040 001C: lw $t3, 8010($gp)
0040 0020: beq $t3, $t2, FFFD
0040 0024: addi $t3, $t3, 0001
0040 0028: bne $t3, $t2, FFFC
0040 002C: syscall

```