

10.1 将内存中 x3000 0000 开始的 5 个 32 位整数分别加上 x3000 0014 开始的 5 个 32 位整数，并存回。

10.2 R1 的初始值的二进制表示含有 7 个 1 时，R2 的最终结果为 7。

11.8 1).

符号	地址
SaveR2	x0000 7000
SaveR4	x0000 7004
HelloWorld	x0000 7008
main	x4000 0000
LOOP	x4000 000C
Return	x4000 0020

2). 该程序实现了输出 Hello, World!, 且将寄存器的值改回了运行程序前的状态。

11.10 1).

x3000 0000	:	NUM1
x3000 0004	:	NUM2
:	:	:
x4000 0000	000001 00000 00001 0000 0000 0000 0000	ADDI R1,R0,#0
x4000 0004	001100 00000 00101 0011 0000 0000 0000	LHI R5,x3000
x4000 0008	011100 00101 00010 0000 0000 0000 0000	LW R2,0(R5)
x4000 000C	001001 00010 00011 0000 0000 0000 0001	ANDI R3,R2,#1
x4000 0010	101001 00011 00000 0000 0000 0000 0100	BNEZ R3,#4
x4000 0014	000011 00010 00010 0000 0000 0000 0001	SUBI R2,R2,#1
x4000 0018	000000 00001 00010 00001 00000 000001	ADD R1,R1,R2
x4000 001C	000011 00010 00010 0000 0000 0000 0010	SUBI R2,R2,#2
x4000 0020	010010 00010 00011 0000 0000 0000 0000	SLEI R3,R2,#0
x4000 0024	101000 00011 00000 1111 1111 1111 0000	BEQZ R3,xFFF0
x4000 0028	001100 00000 00101 0011 0000 0000 0000	LHI R5,x3000
x4000 002C	011101 00101 00001 0000 0000 0000 0100	SW #4(R5),R1
x4000 0030	110000 0000000000000000000000000000	TRAP x00

2). 该程序求小于等于 NUM1 的所有正奇数之和，并存入 NUM2。

11.12 立即数为 16 位补码，最大只能表示 32767，100000 超出了范围，可将

ADDI R1,R0,#100000

拆分为以下两句

ADDI R1,R0,#25000

SLLI R1,R1,#2

汇编时即可发现这一问题。

11.13

```
LB    R5,0(R3)
ADDI  R2,R2,#1
ADDI  R3,R3,#1
ADDI  R1,R0,#1
```

11.14

```
SUBI  R2,R2,#1
LB    R5,0(R3)
ADDI  R3,R3,#1
SUBI  R2,R2,#1
```

11.16

```
(R16:c, R17:x, R18:y)
ADDI  R16,R0,x63
ADDI  R17,R0,#5
```

11.17

```
(R16:i, R17:j, R18:k, R19:x)
      ADDI R8,R0,R16
      BNEZ R16,CASE1
      ADD  R19,R18,R17
      J    EXIT
CASE1: SUBI R16,R16,#1
      BNEZ R16,CASE2
      SUB  R19,R18,R17
      J    EXIT
CASE2: SUBI R16,R16,#1
      BNEZ R16,DEFAULT
      ADDI R19,R18,#2
      J    EXIT
DEFAULT: SUB  R19,R8,R18
EXIT:   TRAP x00
```