Lab 6b executor testcode

Group 1

```
;used to test the instruction of ld,st,add,and
0011 0000 0000 0000;0x3000
0101 000 000 1 00000;and r0,r0,0
0001 000 000 1 00011;add r0,r0,3
0011 000 000000010;st r0,next 3
0010 001 000000001;ld r1
1111 0000 0010 0011;halt
```

```
0011000000000000
0101000000100000
0001000000100011
0011000000000010
0010001000000001
1111000000100101
;you can copy this machine code directly
```

```
result:
R0 = x0003
R1 = x0003
R2 = x7777
R3 = x7777
R4 = x7777
R5 = x7777
R6 = x7777
R7 = x7777
```

Group 2

```
;;used to test the insruction of lea,str,ldr,and,st,ld
0011 0000 0000 0000;0x3000
0101 100 100 1 00000;and r4,r4,#0 r4=0
1110 000 000000000;lea r0,next
0011 000 111111111;st r0,#0
0010 001 111111110;ld r1,#-1
0010 010 000000100;ld r2,next 4 r2=0x3000
0110 011 010 000000;ldr r3,r2,0 r3=[0x30000]
0111 100 010 000000;str r4,r2,0
0110 101 010 000000;ldr r5,r2,#0 r5=0
1111 0000 0010 0101;halt
0011 0000 0000 0000;x3000
```

```
0011000000000000
0101100100100000
1110000000000000
0011000111111111
0010001111111110
0010010000000100
0110011010000000
0111100010000000
0110101010000000
1111000000100101
0011000000000000
;you can copy this machine code directly
```

```
result:
R0 = x3002
R1 = x3002
R2 = x3000
R3 = x5920
R4 = x0000
R5 = x0000
R6 = x7777
R7 = x7777
```

Group 3

```
;used to test the instruction of ldi,sti,and
0011 0000 0000 0000;0x3000
0101 001 001 1 00000;and r1,r1,#0
1010 000 000000011;ldi r0,3 mem[mem[pc+offset]]=mem[x3000]
1011 001 000000010;sti r1,2
1010 010 000000001;ldi r2,1
1111 0000 0010 0101;halt
0011 0000 0000 0000;x3000
```

```
0011000000000000
0101001001100000
1010000000000011
1011001000000010
1010010000000001
1111000000100101
0011000000000000
;you can copy this machine code directly
```

```
result:
R0 = x5260
R1 = x0000
R2 = x0000
R3 = x7777
R4 = x7777
R5 = x7777
R6 = x7777
R7 = x7777
```

Group 4

```
;used to test not
0011 0000 0000 0000;x3000
0101 000 000 1 00000;and r0,r0,#0
0001 000 000 1 00010;add r0,r0,#2
1001 001 000 111111;not r1,r0
1111 0000 0010 0101;halt
```

```
0011000000000000
0101000000100000
0001000000100010
1001001000111111
1111000000100101
;you can copy this machine code directly
```

```
result:
R0 = x0002
R1 = xFFFD
R2 = x7777
R3 = x7777
R4 = x7777
R5 = x7777
R6 = x7777
R7 = x7777
```

Group 5

```
;used to test jsr,jmp
0011000000000000;x3000
0100100000000001;jsr pc+1
0011000000000000;x3000
0101000000100000;and r0,r0,#0
0001000000100010;add r0,r0,#2 r0=2
1110001111111011;lea r1 r1=x3000
0001001001101000;add r1,8 r1=x3008
1100000001000000;jmp r1
0011000000000000;x3000
0001010001100001;add r2,r1,#1 r2=x3009
1111000000100101;halt
```

```
0011000000000000
0100100000000001
0011000000000000
0101000000100000
0001000000100010
1110001111111011
0001001001101000
1100000001000000
0011000000000000
0001010001100001
1111000000100101
;you can copy this machine code directly
```

```
result:
R0 = x0002
R1 = x3008
R2 = x3009
R3 = x7777
R4 = x7777
R5 = x7777
R6 = x7777
R7 = x3001
```

Group 6

```
;used to test the instruction of brz,brp,brn,brzp,brnz,brnzp,brnp
.ORIG x3000
and   x0,x0,#0
brz   label1
add   x1,x1,#1
label1 add x0,x0,#2
brp   label2
add   x2,x2,#2
label2 add x0,x0,#-3
brnz  label3
add   x3,x3,#3
label3 add x0,x0,#2
brnzp label4
add   x4,x4,#4
label4 add x0,x0,#4
brzp  label5
```

```
add   x5,x5,#5
label5 add x0,x0,#-1
brnp  label6
add   x6,x6,#6
label6 trap x25
```

```
0011000000000000
0101000000100000
0000010000000001
0001001001100001
0001000000100010
0000001000000001
0001010010100010
0001000000111101
0000110000000001
0001011011100011
0001000000100010
0000111000000001
0001100100100100
0001000000100100
0000011000000001
0001101101100101
0001000000111111
0000101000000001
0001110110100110
1111000000100101
;you can copy this machine code directly
```

```
result:
R0 = x0004
R1 = x7777
R2 = x7777
R3 = x7777
R4 = x7777
R5 = x7777
R6 = x7777
R7 = x7777
```

☆☆***Group 7***

this code is the original code from the ***PTA oj_test point B3-BR***

```
1000000000000000;x8000
0101 0000 0010 0000;and x0,x0,#0
0101001001100000;and x1,x1,#0
0101010010100000;and x2,x2,#0
0101011011100000;and x3,x3,#0
0101100100100000;and x4,x4,#0
0101101101100000;and x5,x5,#0
0101110110100000;and x6,x6,#0
0101111111100000;and x7,x7,#0
0010 000 000110000;ld x0,#48 x0=x8000
0000001000000010;brp,#2
0000100000000010;brn #2
0001001001100001;add x1,x1,#1
0001100100100001;add x4,x4,#1
0001111111100001;add x7,x7,#1
0110000000000001;ldr x0,x0,#1 x0=0x5260
0000010000000010;brz #2
0000001000000010;brp #2
0001010010100001
0001110110100001
0001011011100001;add x3,x3,#1
1110 000 000100100;lea r0,#36 r0=0x8039
0000001000000010;brp #2
0000100000000010
0001011011100001
0001001001100001;add r1,r1,#1
0001110110100001;add r6,r6,#1
1010000 000011110;ldi r0,#30 0x8000 r0=0x5020
0000110000000010;brnz #2
0000101000000010;brnp #2
0001100100100001
0001011011100001
0001010010100001;add x2,x2,#1
0011 000 000011001;st x0,#25
0000011000000010;brzp #2
0000110000000010
0001101101100001
0001110110100001;add x6,x6,#1
0001111111100001;add x7,x7,#1
0010000000010100;ld x0,#20 x0=0x7FFF
0001000000100010;add x0,x0,#2 x0=8001
0000101000000010;brnp #2
0000111000000010
```

```
0001110110100001
0001001001100001;add x1,x1,#1
0001011011100001;add x3,x3,#1
0000111000000010;brnzp #2
0001111111100001
0001011011100001
0001110110100001;add x6,x6,#1
0010000000000111;ld x0,#7 x0=0x8000
0101 000 000 111111;and x0,x0,1111 1111 1111 1111 x0=8000
0000001000000010;brp #2
0000100000000010;brn #2
0001001001100001
0001101101100001
0001010010100001;add x2,x2,#1
1111000000100101;halt
1000000000000000;0x8000
0000000000000000;->x0=0101000000100000
0111111111111111
;add some explanation in asm
```

```
1000000000000000
0101000000100000
0101001001100000
0101010010100000
0101011011100000
0101100100100000
0101101101100000
0101110110100000
0101111111100000
0010000000110000
0000001000000010
0000100000000010
0001001001100001
0001100100100001
0001111111100001
0110000000000001
0000010000000010
0000001000000010
0001010010100001
0001110110100001
0001011011100001
1110000000100100
```

```
0000001000000010
0000100000000010
0001011011100001
0001001001100001
0001110110100001
1010000000011110
0000110000000010
0000101000000010
0001100100100001
0001011011100001
0001010010100001
0011000000011001
0000011000000010
0000110000000010
0001101101100001
0001110110100001
0001111111100001
0010000000010100
0001000000100010
0000101000000010
0000111000000010
0001110110100001
0001001001100001
0001011011100001
0000111000000010
0001111111100001
0001011011100001
0001110110100001
0010000000000111
0101000000111111
0000001000000010
0000100000000010
0001001001100001
0001101101100001
0001010010100001
1111000000100101
1000000000000000
0000000000000000
0111111111111111
;you can copy this machine code directly
```

```
result:
R0 = x8000
R1 = x0002
R2 = x0002
R3 = x0002
R4 = x0002
R3 = x0002
R4 = x0000
R5 = x0000
R6 = x0003
R7 = x0002
```