HW6

T1

(a) 最后栈中剩：[A, F]
(b) 在 push d 或者 push e 后，栈中元素最多
(c) 如果修改成队列后，最后队列中剩：[E, F]，元素最多的时刻不变

T2

- 因为是中断服务，所以 R7 不会保存返回地址，RET 会跳转到错误的位置
- 直接调用 RET 没有恢复保存下来的 Processor Status Register，因此程序没有从特权模式恢复到用户模式，之前保存的状态码也没有恢复
- RET 不涉及对栈的操作，但从中断服务中恢复，应该要恢复栈指针

T3

(a) 程序可能反复读取同一个字符
(b) KBDR 存的数据可能还没有被读取，用户输入字符可能丢失
(c) display hardware 并不会写入 DDR

T4

统计 x4000 开始的十个数据中首位为1（负数）的个数，存在地址 x5000 处

T5

```
1               .ORIG x3000
2               LEA   R6, STACKBASE
3               LEA   R0, PROMPT
4               TRAP  x22
5               AND   R1, R1, #0
6   LOOP        TRAP  x20
7               TRAP  x21
8               ADD   R3, R0, #-10
9               BRz   INPUTDONE
10              JSR   PUSH
11              ADD   R1, R1, #1
12              BRnzp LOOP
13  INPUTDONE   ADD   R1, R1, #0
```

```
            BRz   DONE
LOOP2       JSR   POP
            TRAP  x21
            ADD   R1, R1, #-1
            BRp   LOOP2
DONE        TRAP  x25

PUSH        ADD   R6, R6, #-2
            STR   R0, R6, #0
            RET
POP         LDR   R0, R6, #0
            ADD   R6, R6, #2
            RET
PROMPT      .STRINGZ "Please enter a sentence:"
            STACKSPAC .BLKW #50
            STACKBASE .FILL #0
            .END
```

R1 寄存的作用是记录输入字符串的长度，便于后续逆序输出

T6

```
          .ORIG x3000
          LD  R6, SP_INIT
          LD  R0, LABELH
          ADD R6, R6, #-1
          STR R0, R6, #0
          LD  R0, LABEL6
          ADD R6, R6, #-1
          STR R0, R6, #0
          LD  R0, LABEL0
          ADD R6, R6, #-1
          STR R0, R6, #0
          LD  R0, LABEL3
          ADD R6, R6, #-1
          STR R0, R6, #0
          RET
LABELH    .FILL x3012
LABEL6    .FILL x5018
LABEL0    .FILL x5000
LABEL3    .FILL x500c
          HALT
SP_INIT   .FILL xFE00
          .END
```

代码的逻辑是把需要调用的子程序地址预先存入栈中，RET直接依次调用，注意 LABELH 的值是代码中 HALT 指令的地址，直接从 x3000 数行数得来

T7

(a) 注意 x10 是十进制的 16 所以循环次数是16次，开局三条指令+最后一次跳转：$8 \times 16 + 3 + 1 = 132$
(b) $144 \times 16 + 3 \times 27 + 16 = 2401$

指令	指令周期
LD	27
ADD	15
LDR	27
STR	26
BR (跳转)	16
BR (不跳转)	15

循环期间：1 个不跳转的 BR，1个跳转的 BR，4 个 ADD，1 个 LDR，1 个 SDR
则一次循环的指令数：$16 + 15 + 4 \times 15 + 27 + 26 = 144$