

HW06

1.

- 共256条trap指令。由于陷入矢量的存放空间为x0000-x00FF，每条指令占用一个地址单元，所以一共可以有256条。
- 由于RET指令执行前用R7保存了下一条指令的地址值用于返回，而BR指令虽然可以强制跳转，但需要提供其跳转地址值，不具有普适性，无法适应在程序中各处调用TRAP指令的需求。
- 对于每个需要保存现场的寄存器值，由于使用的是直接寻址模式，ST和LD各需访问一次内存，所以保存和回复现场要用到寄存器数 * 2 次内存访问。

如果TRAP指令中有其他功能，则额外需要直接寻址次数 * 1 + 间接寻址次数 * 2 次访问内存。

2.

last in first out(LIFO).

一种是通过移动指向栈顶的栈指针实现pop和push操作，一种是通过逐个移动栈中元素实现pop和push操作。其中栈指针操作简便，性能较好；而逐个移动操作极为繁琐，时间复杂度很高，但具体存储形式的可读性较好。

3.

a.

push-push-pop-push-pop-push-push-pop-push-pop-pop-pop-push-push-pop-push-pop-pop

b.

一共存在14种不同的输出方式。

4.

```
POP      ST  R2,SAVER2
          ST  R3,SAVER3
          LD  R2,BASE
          ADD R3,R6,R2
          BRZ FAIL           ;stack is empty
          LDR R1,R6,#0
          ADD R6,R6,#1
          ADD R3,R6,R2
          BRZ FAIL           ;stack is empty
          LDR R0,R6,#0
          ADD R6,R6,#1
          BR  SUCC
PUSH     ST  R2,SAVER2
          ST  R3,SAVER3
          LD  R1,MAX
          ADD R2,R6,R1
          BR  FAIL           ;stack is full
          ADD R6,R6,#-1
          STR R0,R6,#0
          ADD R2,R6,R1
          BR  FAIL           ;stack is full
          ADD R6,R6,#-1
          STR R1,R6,#0
```

```

FAIL    LD    R2,SAVER2
        LD    R3,SAVER3
        AND   R5,R5,#0
        ADD   R5,R5,#1
        RET
SUCC    LD    R2,SAVER2
        LD    R3,SAVER3
        AND   R5,R5,#0
        RET
BASE    .FILL  xC000          ; -x4000
MAX     .FILL  xC010
SAVER2  .FILL  x0000
SAVER3  .FILL  x0000

```

5.

从屏幕输出"EE306 and tests are awesome",并将B处起代码以存一条隔一条的形式将前七条存放在x0025起连续的七个地址空间中。

6.

- A处执行次数等于字符串长度。
- 执行过程中调用了多次TRAP指令，R7值被改变，因此执行结束后调用RET并不能返回原下一条指令地址处

修改后如下。

```

        .ORIG x020F
        ST    R7,SAVER7
START    LDR    R1,R0,#0
        BRZ   DONE
        ST    R0,SAVER0
        ADD   R0,R1,#0
        TRAP  x21
        LD    R0,SAVER0
A        ADD   R0,R0,#1
        BRnzp START
DONE     LD    R7,SAVER7
        RET
SAVER0   .BLKW  #1
SAVER7   .BLKW  #1
        .END

```

7.

a.

最少需要4个栈空间。

$$u = (x * y + w - z) / v$$

b.

PUSH C
PUSH A
ADD
PUSH A
PUSH C
PUSH B
SUB
PUSH D
ADD
MUL
DIV