Homework 6 参考答案

一：

1. 每 6 个 FLOP，读 4 个浮点数，写 2 个浮点数，共访问 24 个字节。
   运算密度 6/(6 * 4) = 0.25
2. 将长度为 300 的向量分解为长度为 44 和 4 个长度 64 的向量。第一次计算长度为 44 的向量，之后长度均为 64。

```
        li          $VL,44          # perform the first 44 ops
        li          $r1,0           # initialize index
loop:   lv          $v1,a_re+$r1    # load a_re
        lv          $v3,b_re+$r1    # load b_re
        mulvv.s     $v5,$v1,$v3     # a+re*b_re
        lv          $v2,a_im+$r1    # load a_im
        lv          $v4,b_im+$r1    # load b_im
        mulvv.s     $v6,$v2,$v4     # a+im*b_im
        subvv.s     $v5,$v5,$v6     # a+re*b_re - a+im*b_im
        sv          $v5,c_re+$r1    # store c_re
        mulvv.s     $v5,$v1,$v4     # a+re*b_im
        mulvv.s     $v6,$v2,$v3     # a+im*b_re
        addvv.s     $v5,$v5,$v6     # a+re*b_im + a+im*b_re
        sv          $v5,c_im+$r1    # store c_im
        bne         $r1,0,else      # check if first iteration
        addi        $r1,$r1,#44     # first iteration,
                                    increment by 44
        j loop                      # guaranteed next iteration
else:   addi        $r1,$r1,#256    # not first iteration,
                                    increment by 256
skip:   blt         $r1,1200,loop   # next iteration?
```

3.

```
1.      mulvv.s     lv          # a_re * b_re (assume already
                                # loaded), load a_im
2.      lv          mulvv.s     # load b_im, a_im*b_im
3.      subvv.s     sv          # subtract and store c _re
4.      mulvv.s     lv          # a_re*b_im, load next a_re vector
5.      mulvv.s     lv          # a_im*b_re, load next b_re vector
6.      addvv.s     sv          # add and store c_im
```

6 chimes

本题中的T_{start}计算较为复杂，不算分。套用公式

$$\lceil \frac{n}{MVL} \rceil (T_{loop} + T_{start}) + n * T_{chime}$$

链接情况下，一个 convey 的多条指令，如果存在相关则该 convey 启动时间为指令启动

时间之和，如果不存在相关则为指令启动时间中取最大值。

4.

```
1. mulvv.s                      # a_re*b_re
2. mulvv.s                      # a_im*b_im
3. subvv.s  sv                  # subtract and store c_re
4. mulvv.s                      # a_re*b_im
5. mulvv.s  lv                  # a_im*b_re, load next a_re
6. addvv.s  sv   lv   lv   lv   # add, store c_im, load next b_re,a_im,b_im
```

二．

1. 1.5 * 16 * 16 = 384 GFLOPS/s

2. 维持吞吐量需要 12 bytes/FLOP × 384 GFLOPs/s = 4.6 TB/s 带宽，给定的存储器带宽不满足，所以吞吐量不能持续