第五章作业

5.15

循环展开如下：

```
void inner4(vec_ptr u, vec_ptr v, data_t *dest) {
  long i;
  long length = vec_length(u);
  data_t *udata = get_vec_start(u);
  data_t *vdata = get_vec_start(v);
  data_t sum = (data_t) 0;
  data_t sum1 = (data_t) 0;
  data_t sum2 = (data_t) 0;
  data_t sum3 = (data_t) 0;
  data_t sum4 = (data_t) 0;
  data_t sum5 = (data_t) 0;

  for (i = 0; i < length-6; i+=6) {
    sum = sum + udata[i] * vdata[i];
    sum1 = sum1 + udata[i+1] * vdata[i+1];
    sum2 = sum2 + udata[i+2] * vdata[i+2];
    sum3 = sum3 + udata[i+3] * vdata[i+3];
    sum4 = sum4 + udata[i+4] * vdata[i+4];
    sum5 = sum5 + udata[i+5] * vdata[i+5];
  }
  for(; i < length; ++i)
    sum = sum + udata[i] * vdata[i];
  *dest = sum + sum1 + sum2 + sum3 + sum4 + sum5;
}
```

处理器的浮点数乘法的容量只有 2（<6），限制了 CPE。

5.19

```
void psum_4_1a(float a[], float p[], long n) {
  long i;
  float val, last_val;
  float tmp, tmp1, tmp2, tmp3;
  last_val = p[0] = a[0];

  for (i = 1; i < n - 4; i++) {
    tmp = last_val + a[i];
    tmp1 = tmp + a[i+1];
```

```
    tmp2 = tmp1 + a[i+2];
    tmp3 = tmp2 + a[i+3];

    p[i] = tmp;
    p[i+1] = tmp1;
    p[i+2] = tmp2;
    p[i+3] = tmp3;

    /* key point */
    last_val = last_val + (a[i] + a[i+1] + a[i+2] + a[i+3]);
  }

  for (; i < n; i++) {
    last_val += a[i];
    p[i] = last_val;
  }
}
```