

#### 4.46

A. 没有正确描述指令 `popq %rsp`。该段代码序列使 `%rsp` 的值为 `(%rsp)+8`，而指令 `popq %rsp` 执行后，`%rsp` 的值应为 `(%rsp)`。

B. 改写代码序列：

```
addq $8 , %rsp
movq -8(%rsp) , REG
```

#### 4.51

阶段	iaddq V, rB
取址	<code>icode:ifun &lt;- M1 [PC]</code> <code>rA:rB &lt;- M1[PC+1]</code> <code>valC &lt;- M8[PC+2]</code> <code>valP &lt;- PC+10</code>
译码	<code>valB &lt;- R[rB]</code>
执行	<code>valE &lt;- valC + valB</code>
访存	无
写回	<code>R[rB] &lt;- valE</code>
更新 PC	<code>PC &lt;- valP</code>

#### 5.14

```
/* Inner product. Accumulate in temporary */
void inner5(vec_ptr u, vec_ptr v, data_t* dest) {
    long i;
    long length = vec_length(u);
    long limit = length - 5;
    data_t* udata = get_ver_start(u);
    data_t* vdata = get_vec_start(v);
    data_t sum = (data_t)0;
    for (i = 0; i < limit; i += 6) {
        sum = sum + udata[i] * vdata[i] +
            udata[i + 1] * vdata[i + 1] +
            udata[i + 2] * vdata[i + 2] +
            udata[i + 3] * vdata[i + 3] +
            udata[i + 4] * vdata[i + 4] +
            udata[i + 5] * vdata[i + 5];
    }
    for (; i < length; i++) {
        sum = sum + udata[i] * vdata[i];
    }
    *dest = sum;
}
```