```
/* Compute i,k of fixed matrix product */
int fix_prod_ele (fix_matrix A, fix_matrix B, long i, long k) {
    long j;
    int result = 0;

    for (j = 0; j < N; j++)
        result += A[i][j] * B[j][k];

    return result;
}
```

a）原始的C代码

```
1   /* Compute i,k of fixed matrix product */
2   int fix_prod_ele_opt(fix_matrix A, fix_matrix B, long i, long k) {
3       int *Aptr = &A[i][0];    /* Points to elements in row i of A    */
4       int *Bptr = &B[0][k];    /* Points to elements in column k of B */
5       int *Bend = &B[N][k];    /* Marks stopping point for Bptr       */
6       int result = 0;
7       do {                          /* No need for initial test */
8           result += *Aptr * *Bptr;  /* Add next product to sum  */
9           Aptr ++;                  /* Move Aptr to next column */
10          Bptr += N;                /* Move Bptr to next row    */
11      } while (Bptr != Bend);       /* Test for stopping point  */
12      return result;
13  }
```

b）优化过的C代码

图 3-37 原始的和优化过的代码，该代码计算定长数组的矩阵乘积的元素 i，k。编译器会自动完成这些优化

```
    int fix_prod_ele_opt(fix_matrix A, fix_matrix B, long i, long k)
    A in %rdi, B in %rsi, i in %rdx, k in %rcx
1   fix_prod_ele:
2     salq    $6, %rdx              Compute 64 * i
3     addq    %rdx, %rdi            Compute Aptr = xA + 64i = &A[i][0]
4     leaq    (%rsi,%rcx,4), %rcx   Compute Bptr = xB + 4k = &B[0][k]
5     leaq    1024(%rcx), %rsi      Compute Bend = xB + 4k + 1024 = &B[N][k]
6     movl    $0, %eax              Set result = 0
7   .L7:                          loop:
8     movl    (%rdi), %edx          Read *Aptr
9     imull   (%rcx), %edx          Multiply by *Bptr
10    addl    %edx, %eax            Add to result
11    addq    $4, %rdi              Increment Aptr ++
12    addq    $64, %rcx             Increment Bptr += N
13    cmpq    %rsi, %rcx            Compare Bptr:Bend
14    jne     .L7                   If !=, goto loop
15    rep; ret                      Return
```

练习题 3.39 利用等式 3.1 来解释图 3-37b 的 C 代码中 Aptr、Bptr 和 Bend 的初始值计算(第 3～5 行)是如何正确反映 fix_prod_ele 的汇编代码中它们的计算(第 3～5 行)的。