

```

// Function to compute the inner product of two vectors
// This version uses 6x6 loop unrolling
void inner6x6(vec_ptr u, vec_ptr v, data_t *dest){
    // Declare index variable
    long i;

    // Get the length of the vector u
    long length = vec_length(u);

    // Get the start of the vectors u and v
    data_t *udata = get_vec_start(u);
    data_t *vdata = get_vec_start(v);

    // Initialize six sum variables to 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;
    data_t sum6 = (data_t) 0;

    // Loop over the vectors in chunks of 6
    for(i = 0; i < length-5; i+=6){
        // Compute the sum of the products of the corresponding elements in u and v
        sum1 = sum1 + (udata[i] * vdata[i]);
        sum2 = sum2 + (udata[i+1] * vdata[i+1]);
        sum3 = sum3 + (udata[i+2] * vdata[i+2]);
        sum4 = sum4 + (udata[i+3] * vdata[i+3]);
        sum5 = sum5 + (udata[i+4] * vdata[i+4]);
        sum6 = sum6 + (udata[i+5] * vdata[i+5]);
    }

    // Add all the sums together and store the result in dest
    *dest = (sum1 + sum2) + (sum3 + sum4) + (sum5 + sum6);

    // for vectors whose length is not a multiple of 6
    // compute any remaining elements
    for(; i < length; i++){
        *dest += udata[i] * vdata[i];
    }
}

```

