```
CAPA Report

Summary: TotalFiles=1 Files With Improvements=1

/home/james/Projects/LatexDocs/DesignDoc/Code/CodeUnderTest.cpp:22:1
Pattern: Vectorisable Function Declaration Priority: 3 Info: Function Declaration
void reshape2vec(float * out_vec, float ** in_mat, size_t dim);

/home/james/Projects/LatexDocs/DesignDoc/Code/CodeUnderTest.cpp:76:1
Pattern: Vectorisable Function Declaration Priority: 3 Info: Function Declaration
void mmult(float ** A, float ** B, float ** C, size_t dim);

/home/james/Projects/LatexDocs/DesignDoc/Code/CodeUnderTest.cpp:42:5
Pattern: Map Priority: 3 Info: Stride Size: 1. Number of Elements: 1000000.
Potential Speedup: 158.03 ~ 140.51
for (size_t i = 0; i < ELEMS; ++i){
        starting_vec[i] /= 2;
        starting_vec[i] += 4;
    }

/home/james/Projects/LatexDocs/DesignDoc/Code/CodeUnderTest.cpp:49:5
Pattern: Reduce Priority: 2 Info: Stride Size: 1. Number of Elements: 1000000.
Potential Speedup: 30.30 ~ 34.33
for (size_t i = 0; i < ELEMS; ++i)
        k += starting_vec[i]/ELEMS

/home/james/Projects/LatexDocs/DesignDoc/Code/CodeUnderTest.cpp:55:5
Pattern: Scan Priority: 2 Info: Stride Size: 1. Number of Elements: 1000000.
Potential Speedup: 19.72 ~ 30.11
for (size_t i = 1; i < ELEMS; i++)
        cum_sum[i] = starting_vec[i]/ELEMS + cum_sum[i-1]

/home/james/Projects/LatexDocs/DesignDoc/Code/CodeUnderTest.cpp:77:5
Pattern: Matrix Multiplication Priority: 1 Info: A Matrix Multiply
Potential Speedup: 16.72 ~ 229.00
for (size_t i = 0; i < dim; ++i)
        for (size_t j = 0; j < dim; ++i)
            for (size_t k = 0; k < dim; ++i)
                C[i][j] += A[i][k] * B[k][j]

/home/james/Projects/LatexDocs/DesignDoc/Code/CodeUnderTest.cpp:23:5
Pattern: Vectorisable region Priority: 5 Info: Generally vectorisable region of code
for (size_t i = 0; i < dim; ++i)
        for (size_t j = 0; j < dim; ++j)
            out_vec[i*dim + j] = in_mat[i][j]


[CAPA v0.10.2]
```