```
from chill import *

source('mm.c')
procedure('mm')
loop(0)
known('ambn > 0')
known('an > 0')
known('bm > 0')
print_code()
permute([1], [3,1,2])
print_code()

void mm(float **A, float **B, float **C, int ambn, int an, int bm) {
    int i, j, n;

    for(i = 0; i < an; i++) {
        for(j = 0; j < bm; j++) {
            C[i][j] = 0.0f;
            for(n = 0; n < ambn; n++) {
                C[i][j] += A[i][n] * B[n][j];
            }
        }
    }
}

void mm(float **A,float **B,float **C,int ambn,int an,int bm)
{
    int t6;
    int t4;
    int t2;
    int i;
    int j;
    int n;
    for (t2 = 0; t2 <= an - 1; t2 += 1) {                /* i(1) */
        for (t4 = 0; t4 <= bm - 1; t4 += 1) {            /* j(2) */

            if (t2 + 1 <= ambn) {                        /* if i < ambn */
                C[t2][t4] = 0.0f;
                if (t4 + 1 <= an)
                    C[t4][0] += (A[t4][t2] * B[t2][0]);
            } else
                C[t2][t4] = 0.0f;

            if (t2 + 1 <= ambn && t4 + 1 <= an)
                for (t6 = 1; t6 <= bm - 1; t6 += 1)
                    C[t4][t6] += (A[t4][t2] * B[t2][t6]);
        }
        if (t2 + 1 <= ambn)
            for (t4 = bm; t4 <= an - 1; t4 += 1)         /*  */
                for (t6 = 0; t6 <= bm - 1; t6 += 1)
                    C[t4][t6] += (A[t4][t2] * B[t2][t6]);
    }

    for (t2 = an; t2 <= ambn - 1; t2 += 1)               /* n(3) */
        for (t4 = 0; t4 <= an - 1; t4 += 1)              /* i(1) */
            for (t6 = 0; t6 <= bm - 1; t6 += 1)          /* j(2) */
                C[t4][t6] += (A[t4][t2] * B[t2][t6]);
}
```