```c
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

void arrTest(int, FILE*);

int main() {
    FILE *f;
    int m;
    f = fopen("memory-frag-output.txt", "w");

    printf("\nEnter a value for m to go to: ");
    scanf("%d", &m);

    fprintf(f, "m, Step 1, Step 2\n");
/*     for(int i = 1000; i < m; i += 1000)
        arrTest(i, f); */
    arrTest(m, f);
}

void arrTest(int m, FILE * f) {
    double time_spent;

    printf("m is %d\t\t", m);
    fprintf(f, "%d", m);

    // Step 1
    clock_t begin_step1 = clock();
    // allocate memory for 3m int arrays for size 800000 each
    int** arrays1 = malloc(sizeof(int*) * 3*m);
    for(int i = 0; i < 3*m; i++)
        arrays1[i] = malloc(sizeof(int[800000]));
    clock_t end_step1 = clock();
    time_spent = (double)(end_step1-begin_step1) / CLOCKS_PER_SEC;
    printf(" | Step 1: %f seconds", time_spent);
    fprintf(f, " %f,", time_spent);

    // Step 2
    clock_t begin_step2 = clock();
    // deallocate memory even numbered arrays from 3m arrays
    for(int i = 0; i < 3*m; i += 2)
        free(arrays1[i]);
    // and allocate memory for m int arrays for size 9000000 each
    int** arrays2 = malloc(sizeof(int*) * m);
    for(int i = 0; i < m; i++)
        arrays2[i] = malloc(sizeof(int[900000]));
    clock_t end_step2 = clock();
    time_spent = (double)(end_step2-begin_step2) / CLOCKS_PER_SEC;
    printf(" | Step 2: %f seconds\n", time_spent);
    fprintf(f, "%f\n", time_spent);

    free(arrays1);
    free(arrays2);
}

```