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

int searchAndReportPositive(int **image, int **sub, int i0, int j0, int p, int q){
    int i, j;
    for(i = 0; i < p; i++)
        for(j = 0; j < q; j++)
            if(image[i0 + i][j0 + j] != sub[i][j])
                return 0;
    printf("POSITIVE MATCH AT (%d, %d)\n", i0, j0);
    return 1;
}

int searchAndReportNegative(int **image, int **sub, int i0, int j0, int p, int q){
    int i, j;
    for(i = 0; i < p; i++)
        for(j = 0; j < q; j++)
            if(image[i0 + i][j0 + j] != (1 - sub[i][j]))
                return 0;
    printf("NEGATIVE MATCH AT (%d, %d)\n", i0, j0);
    return 1;
}

int main(){
    int m, n, p, q, i, j, count = 0;
    scanf("%d %d", &m, &n);
    scanf("%d %d", &p, &q);

    int **image = (int**)malloc(m * sizeof(int*));
    int **sub = (int**)malloc(p * sizeof(int*));

    // Read the matrix and the submatrix
    for(i = 0; i < m; i++){
        image[i] = (int*)malloc(n * sizeof(int));
        for(j = 0; j < n; j++)
            scanf("%d", &image[i][j]);
    }
    for(i = 0; i < p; i++){
        sub[i] = (int*)malloc(q * sizeof(int));
        for(j = 0; j < q; j++)
            scanf("%d", &sub[i][j]);
    }

    // Be careful about limits of the counters i and j
    // Do not read past the row and column limits
    for(i = 0; i + p - 1 < m; i++){
        for(j = 0; j + q - 1 < n; j++){
            // Check if the submatrix appears as is at index (i,j)
            count += searchAndReportPositive(image, sub, i, j, p, q);
            // Check if the complemented submatrix appears at index (i,j)
            count += searchAndReportNegative(image, sub, i, j, p, q);
        }
    }

    if(count == 0)
        printf("NO MATCHES");
    return 0;
}
```