Path in Matrix

> 给定NxN矩阵，求从第一行任意列开始到最后一行任意列的最大路径和——要求每次移动时，只能从(i,j)移动到(i + 1, j)、(i + 1, j - 1)或者(i+ 1, j+1)

```
#include <bits/stdc++.h>

using namespace std;

typedef vector<int> vi;
typedef vector<vector<int>> vvi;

int maxSumPath(vvi &matrix, int &N) {
    vvi dp(N, vi(N));
    for (int j = 0; j < N; ++j) dp[0][j] = matrix[0][j];
    for (int i = 1; i < N; ++i) {
        for (int j = 0; j < N; ++j) {
            if (j == 0)
                dp[i][j] = max(dp[i - 1][j], dp[i - 1][j + 1]) + matrix[i][j];
            else if (j == N - 1)
                dp[i][j] = max(dp[i - 1][j - 1], dp[i - 1][j]) + matrix[i][j];
            else
                dp[i][j] = max(max(dp[i - 1][j - 1], dp[i - 1][j]), dp[i - 1][j + 1]) + matrix[i][j];
        }
    }
    return *max_element(dp[N - 1].begin(), dp[N - 1].end());
}

int main() {
    int T;
    scanf("%d", &T);
    while (T--) {
        int N;
        scanf("%d", &N);
        vvi matrix(N, vi(N));
        for (int i = 0; i < N; ++i)
            for (int j = 0; j < N; ++j)
                scanf("%d", &matrix[i][j]);
        printf("%d\n", maxSumPath(matrix, N));
    }
}
```