Maximum sum of a path in a Right Number Triangle

> 在右侧三角形中，找到第一层到最后一层的最大路径和

右侧三角形如下：

```
1
2 3
4 5 6
......
```

采用动态规划，从最后一层递推至第一层，状态转移方程为：

$$dp[i][j] = max(dp[i+1][j], dp[i+1][j+1]) + triangle[i][j];$$

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

using namespace std;

int maxSum(vector<vector<int>> &triangle, int &N) {
    vector<vector<int>> dp(N, vector<int>(N));
    for (int j = N - 1; j >= 0; --j) dp[N - 1][j] = triangle[N - 1][j];
    for (int i = N - 2; i >= 0; --i) {
        for (int j = 0; j <= i; ++j) {
            dp[i][j] = max(dp[i + 1][j], dp[i + 1][j + 1]) + triangle[i][j];
        }
    }
    return dp[0][0];
}

int main() {
    int T;
    scanf("%d", &T);
    while (T--) {
        int N;
        scanf("%d", &N);
        vector<vector<int>> triangle(N, vector<int>(N));
        for (int i = 0; i < N; ++i) {
            for (int j = 0; j <= i; ++j) {
                scanf("%d", &triangle[i][j]);
            }
        }
        printf("%d\n", maxSum(triangle, N));
    }
    return 0;
```

```
}
```