Zoj3158Cut The CakeDFS暴力枚举

**题意**：从上往下切割，不能切边缘，然后问你两个差值，要小于给出的值。

**分析**：对于每一行 ，都要从某一位置分成两段 ，由于不能切出0长度的一段，所 以有n - 1个选择 ，一共m行 ，就是(n - 1)m种方案 。由于问题的规模很 小 ，暴力枚举所有情况取最优解就可以了。直接用dfs即可。

#include <iostream>

#include <cmath>

using namespace std;

const int MAXSIZE = 8;

long nutrition[MAXSIZE][MAXSIZE];

int m, n;

long mindiff = -1;

void dfs(int i, long sumleft, long sumright){//当前i行情形

if (i == m){

int sum = abs(sumleft-sumright);

if (sum < mindiff || mindiff == -1){

mindiff = sum;

}

return;

}

//不是最后一行，则逐一分割当前行i，探测下一行

for (int k = 1; k < n; k++){//n-1个切位

long tleft = 0;

long tright = 0;

for (int s = 0; s < k; s++)

tleft += nutrition[i][s];

for (int s = k; s < n; s++)

tright += nutrition[i][s];

dfs(i + 1, sumleft + tleft, sumright + tright);

}

}

int main(){

while (cin >> m >> n){

for (int i = 0; i < m; i++){

for (int j = 0; j < n; j++){

cin >> nutrition[i][j];

}

}

int t;

cin >> t;

mindiff = -1;

//从0行开始，该行有n-1种切位

for (int i = 1; i < n; i++){

long sumleft = 0;

long sumright = 0;

for (int j = 0; j < i; j++)

sumleft += nutrition[0][j];

for (int j = i; j < n; j++)

sumright += nutrition[0][j];

dfs(1, sumleft, sumright);

}

if (mindiff <= t && mindiff != -1)

cout << mindiff << endl;

else

cout << "You'd better buy another one!" << endl;

}

}

…

Main

判断mindiff

Dfs(1)

全局mindiff

Dfs(1)

全局mindiff

Dfs(m)

全局mindiff

Dfs(m)

全局mindiff