**★★★☆☆**

**題組：Problem Set Archive** **with Online Judge**

**題號：12621: On a Diet**

**解題者：楊昌彪**

**解題日期：2024年5月16日**

**題意：**

給定一最少攝入熱量及一些有標示熱量的物品，試求在不能低於最少攝入熱量的條件下，從這些物品能夠攝取到的最小熱量。

**題意範例：**

2140
4

450 150 120 50 => No solution!

**解法：**

此題要求出子集合的所有總和可能，為Sum of subset的題目。使用Dynamic programming來算總和，宣告一整數一維陣列dp[25000]，陣列的值為1 該index能夠被子集合組合出來，0則相反，dp[0] = 1。每一次跑子集合內的一個元素，陣列的index從25000開始往回走訪，若走訪到該index的值為1，說明在跑此元素之前已能將此index組合出來，因此將dp[index+元素之標示熱量]設為1，說明index+元素之標示熱量也能夠被組合出來。

**解法範例：**

2480 => 最小攝入熱量

5 => 集合大小

1230 1050 820 890 1150 => 元素之標示熱量

sol: round = 1 => dp[1230] = 1

 round = 2 => dp[2280] = 1, dp[1230] = 1, dp[1050] = 1

 round = 3 => dp[3100] = 1, dp[2280] = 1, dp[2050] = 1, dp[1870] = 1, dp[1230] = 1, dp[1050] = 1, dp[820] = 1

 round = 4 => dp[3990] = 1, dp[3170] = 1, dp[3100] = 1, dp[2940] = 1, dp[2760] = 1, dp[2280] = 1, dp[2120] = 1, dp[2050] = 1, dp[1940] = 1, dp[1870], dp[1710] = 1, dp[1230] = 1, dp[1050] = 1, dp[890] = 1, dp[820] = 1

**討論：**

(1)走訪dp時，必需從25000開始往1走訪，若從1開始走訪，會有多紀錄的問題。

**程式：**

#include <iostream>

using namespace std;

int dp[25001];

int main()

{

 int t = 0;

 cin >> t;

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

 {

 for (int j = 0; j < 25001; j++)

 {

 dp[j] = 0;

 }

 int cal = 0, c = 0;

 cin >> cal;

 cin >> c;

 cal /= 10;

 int course = 0;

 dp[0] = 1;

 int sum = 0;

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

 {

 cin >> course;

 course /= 10;

 for (int k = 25001; k >= 0; k--)

 {

 if (dp[k] == 1 && k + course < 25001)

 {

 dp[k + course] = 1;

 }

 }

 sum += course;

 }

 if (sum < cal)

 {

 cout << "NO SOLUTION" << endl;

 continue;

 }

 for (int j = cal; j < 25001; j++)

 {

 if (dp[j] == 1)

 {

 cout << j \* 10 << endl;

 break;

 }

 }

 }

}