**Fredo and Game**

Max. Marks: 100

Fredo is playing a game. The rules of the game are:  
Initially, you are given AA units of ammo. There are NN obstacles placed on a path. When you hit an obstacle, you **gain three units ammo** and **lose one unit of ammo**. When you don't hit an obstacle, you **lose one unit ammo**. If at any instance, you are left with 00 ammo units, the game ends there.

Fredo has an array ArrArr containing NN elements corresponding to the NN obstacles. If Fredo will hit obstacle ii , then Arr[i]=1Arr[i]=1 else Arr[i]=0Arr[i]=0.  
Fredo asks you to tell him if he will be able to reach the end of the path. If yes, then also tell him the remaining number of ammo units.  
If he is not able to reach the end of the path, tell him the obstacle index at which his game would end.

**Note**: If Fredo reaches the last obstacle, he is said to reach the end of the path.

**Input Format**:  
The first line consists of an integer TT , denoting the number of test cases.  
Each test cases consists of two lines:  
The first line consists of two integers AA and NN, denoting the initial ammo units Fredo has and the number of obstacles respectively.  
The second line consists of array ArrArr as described in the question.

**Output Format**:  
For each test case:  
If he is able to reach the end of the path, print **Yes** followed by the number of ammo units remaining.  
Else print **No** followed by the index (1 based) of the obstacle at which the game ends.

**Input Constraints**:  
1≤T≤101≤T≤10  
1≤A≤1051≤A≤105  
1≤N≤1051≤N≤105  
0≤Arr[i]≤10≤Arr[i]≤1

**SAMPLE INPUT**

2

5 5

0 0 1 0 1

2 5

0 0 1 0 1

**SAMPLE OUTPUT**

Yes 6

No 2

**Explanation**

Test case 11:  
Initially he has 55 units of ammo.  
at first obstacle, he is left with 44 units of ammo.(he loses one ammo unit at this obstacle)  
at second obstacle, he is left with 33 units of ammo.  
at third obstacle, he is left with 3+3−1=53+3−1=5 units of ammo.(he gains three and loses one ammo unit)  
at fourth obstacle,he is left with 44 ammo units.  
at fifth obstacle, he is left with 66 ammo units.  
Answer: Yes 66

Test case 22:  
The game ends at obstacle 22 as he is left with 00 ammo units there.

**Time Limit:**1.0 sec(s) for each input file.

**Memory Limit:**256 MB

**Source Limit:**1024 KB

**Marking Scheme:**Marks are awarded if any testcase passes.

**Allowed Languages:**C, C++, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Scala 2.11.8, Swift, Visual Basic

<https://www.hackerearth.com/challenge/competitive/april-circuits-17/algorithm/fredo-and-game/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication2

{

class Program

{

static void Main(string[] args)

{

int t = int.Parse(Console.ReadLine());

while (t-- > 0)

{

string[] input = Console.ReadLine().Split(' ');

int ammo = int.Parse(input[0]);

int n = int.Parse(input[1]);

int[] arr = Array.ConvertAll(Console.ReadLine().Split(' '), e => int.Parse(e));

//int ammo = 5;

//int[] arr = { 0, 0, 1, 0, 1 };

//int ammo = 2;

//int[] arr = { 0, 0, 1, 0, 1 };

int i = 0;

for (i = 0; i < arr.Length; i++)

{

if (arr[i] == 0)

{

ammo--;

}

else if (arr[i] == 1)

{

ammo += 2;

}

if (ammo <= 0)

{

break;

}

}

if (i >= arr.Length - 1)

{

Console.WriteLine("Yes " + ammo);

}

else

{

Console.WriteLine("No " + (i + 1));

}

}

Console.ReadLine();

}

}

}