// C++ program to check if there is a subset

// with sum divisible by m.

#include <bits/stdc++.h>

using namespace std;

// Returns true if there is a subset

// of arr[] with sum divisible by m

bool modularSum(int arr[], int n, int m)

{

     bool DP[m];

    memset(DP, false, m);

    // we'll loop through all the elements of arr[]

    for (int i=0; i<n; i++)

    {

        // anytime we encounter a sum divisible

        // by m, we are done

        if (DP[0])

            return true;

        bool temp[m];

        memset(temp,false,m);

        for (int j=0; j<m; j++)

        {

            // if an element is true in DP table

            if (DP[j] == true)

            {

                if (DP[(j+arr[i]) % m] == false)

                    // We update it in temp and update

                    // to DP once loop of j is over

                    temp[(j+arr[i]) % m] = true;

            }

        }

        for (int j=0; j<m; j++)

            if (temp[j])

                DP[j] = true;

        DP[arr[i]%m] = true;

    }

    return DP[0];

}