Programming Biro Assignment 2

Task: More expensive tickets

A travel agency stores how much the plane tickets to cities being different distances away cost. Write a program that lists the cities to which the price per kilometer is greater than HUF 100.

Specification

Input

$N \in \mathbb{N}$

$travel_{1..N} \in travels^{N}$, $travels = (distance \times price)^{*}$

$distance \in \mathbb{N}$

$price \in \mathbb{N}$

$P : \mathbb{N} \to \mathbb{L}$, $P(x) = (travel_x.price / travel_x.distance \geq 100)$

Output

$count \in \mathbb{N}$

$Y \in \mathbb{N}^{count}$

Precondition

$0 \leq N \leq 100$

$\forall i\ (1 \leq i \leq N) : 1 \leq travel_i.distance \leq 20000$

$\forall i\ (1 \leq i \leq N) : 1 \leq travel_i.price \leq 200000$

Postcondition

$$count = \sum_{i=1}^{N} 1$$

$P(i)$ or $(travel_i.price / travel_i.distance \geq 100)$

and $\forall i\ (1 \leq i \leq count) : P(travel[Y[i]])$

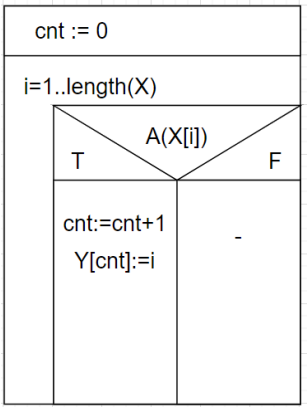
and $Y \subseteq (1, 2, .. length(travel))$

or

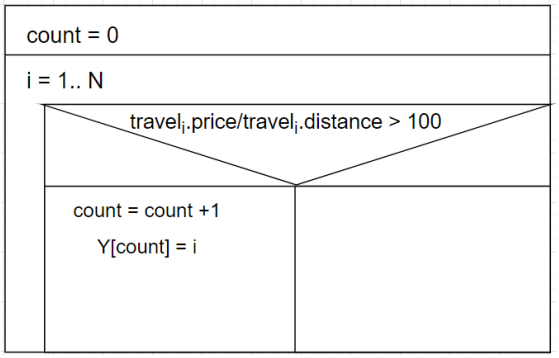
$Y \subseteq (1, 2 .. N)$

Algorithm

Pattern -> Multiple item selection



Pattern		Task
cnt	->	count
length(X)	->	N
A	->	$travel_i.price/travel_i.distance>100$
Y	->	Y



Implementation - Code

```
using System;

namespace ConsoleApp1
{
    struct travels
    {
        public double distance;
        public double price;
    }
    internal class Program
    {
        static void Main(string[] args)
        {
            ///Declaration
            ///
            int N;
```

```
            const int MaxN = 100;
            travels[] travel = new travels[MaxN];
            int count;
            int[] Y;

            ///Input
            ///
            N = Int32.Parse(Console.ReadLine());
            for (int i = 0; i < N; i++)
            {
                string[] tmp = Console.ReadLine().Split(" ");
                travel[i].distance = double.Parse(tmp[0]);
                travel[i].price = double.Parse(tmp[1]);
            }

            ///Algorithm
            ///
            count = 0;
            Y = new int[N];
            for (int i = 0; i < N; i++)
            {
                if ((travel[i].price / travel[i].distance) > 100)
                {
                    Y[count++] = i+1;
                }
            }

            ///Output
            ///
            string output = "";
            for (int i = 0; i < count; i++)
            {
                output = output + " " + Y[i];
            }
            output = count + output;
            Console.WriteLine(output);
        }
    }
}
```