

---

## Algorithmen und Wahrscheinlichkeit

### Programming Exercises 7

---

DEADLINE: 10 AM @ 31.05.2018

#### Exercise 1 – *Santa Claus*

Like every new year, Santa Claus is giving away presents to children. This year he wants to approach the problem in a more systematic way. There are  $n$  children, denoted by  $0, 1, \dots, n-1$ , and Santa estimated for each child how good they were in the current year, which translates into how many toys should they get. Furthermore, Santa has  $m$  different types of toys available, denoted by  $0, \dots, m-1$ , and for each type  $i$  he can get  $c_i$  copies of toys of that type. While it is fine if some of the toys are not given to anybody, Santa wants to make sure that each child receives the amount of toys he/she deserves. Moreover, a single child should not get two toys of the same type. Please, help Santa determine if such a toy allocation is possible.

**Input** The first line of the input file contains an integer  $1 \leq t \leq 20$  denoting the number of test cases that follow. Each of the  $t$  test cases is described as follows.

- It starts with a line containing two integers  $n$   $m$ , separated by space, denoting the number of children and the number different toy types, respectively, such that  $1 \leq n, m \leq 100$ .
- The next line contains  $n$  numbers  $d_0 \dots d_{n-1}$ , where  $d_i$  denotes the number of toys child  $i$  deserves and such that  $0 \leq d_i \leq 10^3$ , for every  $i \in \{0, \dots, n-1\}$ .
- The next line contains  $m$  numbers  $c_0 \dots c_{m-1}$ , where  $c_i$  denotes the number of copies of the toy type  $i$  Santa possesses and such that  $0 \leq c_i \leq 10^3$ , for every  $i \in \{0, \dots, m-1\}$ .

**Output** For each test case output one line containing **yes** if Santa can distribute the toys according to his criteria and **no** otherwise.

**Points** This exercise gives 2 points.

**Hint** You might want to consider using a flow algorithm. We recommend that you do not try to implement a flow algorithm yourself, but to use the implementation provided at [https://www.cadmo.ethz.ch/education/lectures/FS18/AW/restricted/programming\\_exercises/template/template\\_flow.zip](https://www.cadmo.ethz.ch/education/lectures/FS18/AW/restricted/programming_exercises/template/template_flow.zip)

#### Sample Input

```
4
2 2
2 0
0 2
2 2
```

#### Sample Output

```
no
yes
yes
no
```

2 1  
1 2  
5 1  
1 1 1 1 1  
5  
1 6  
6  
1 1 1 1 1 0