HARDIK HIRAMAN PAWAR
(1RV21CS046)

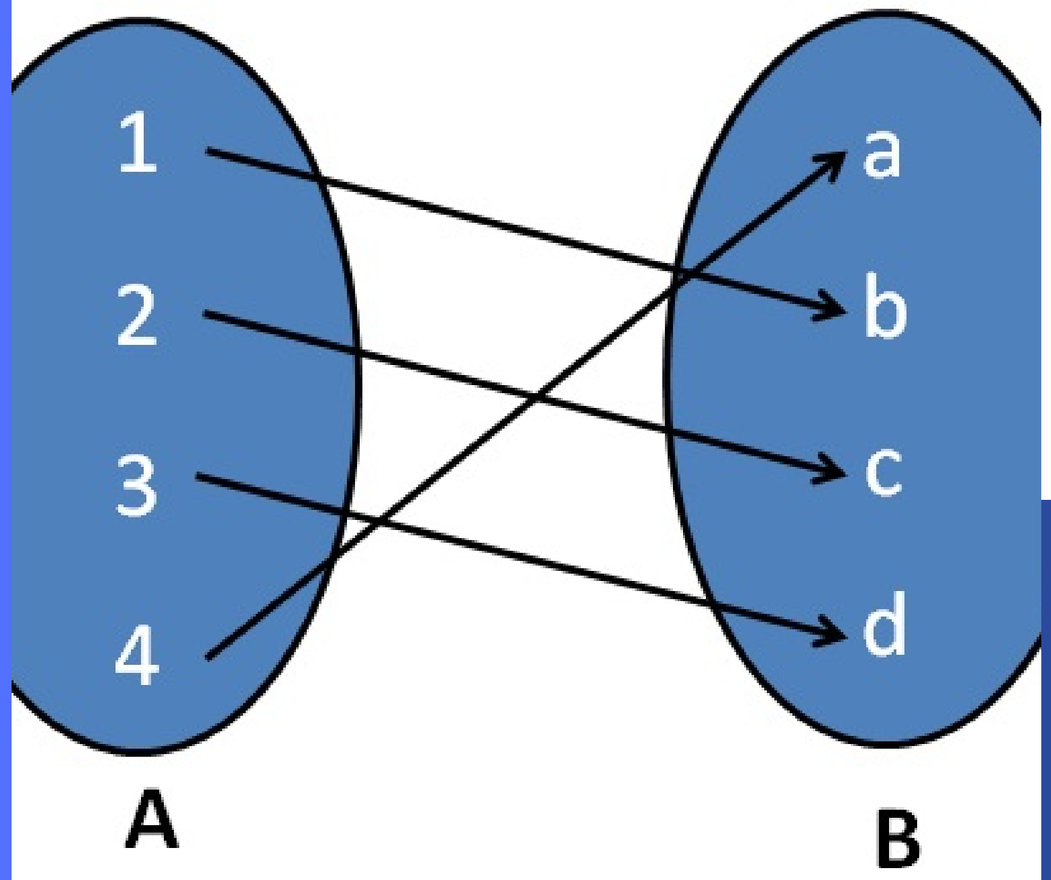
3RD SEM
DMS EL

CALCULATION OF NO. OF ONTO FUNCTIONS

INTRODUCTION

WHAT IS AN ONTO FUNCTION?

teacho



DEFINITION

A SURJECTIVE (ONTO) FUNCTION IS A FUNCTION F SUCH THAT EVERY ELEMENT Y CAN BE MAPPED FROM ELEMENT X SO THAT F(X) = Y.

- Range = Co-Domain
- *Every element of the function's codomain is the image of atleast one element of its domain.*

CODE

PYTHON PROGRAM EXPLANATION

1

FUNCTION DEFINITION

The code defines a function that generates all onto functions from a finite set S to another finite set T.

2

CUSTOMIZATION

*The functionality **product** generates all possible combinations of length **n** of **T.***

3

RETURN VALUE

The function returns the total number of onto functions and a list of all the mappings. An example usage prompts the user for whether to allow repeated elements in the mappings and prints the results.

```python
from itertools import product


def calculate_onto_functions(S, T):

    # Calculate the number of onto functions from S to T
    n = len(S)
    m = len(T)
    num_onto = m ** n

    # List all of the onto functions from S to T
    onto = []

    for t in product(T, repeat=n):
        onto.append(dict(zip(S, t)))

    return num_onto, onto
```

EXAMPLES

Default

FOR THE SETS:

S = {1, 2, 3, 4}

T = {'A', 'B', 'C'}

```
Number of onto functions from S to T: 81
All of the onto functions from S to T:

{1: 'c', 2: 'c', 3: 'c', 4: 'c'}
{1: 'c', 2: 'c', 3: 'c', 4: 'b'}
{1: 'c', 2: 'c', 3: 'c', 4: 'a'}
{1: 'c', 2: 'c', 3: 'b', 4: 'c'}
{1: 'c', 2: 'c', 3: 'b', 4: 'b'}
{1: 'c', 2: 'c', 3: 'b', 4: 'a'}
{1: 'c', 2: 'c', 3: 'a', 4: 'c'}
{1: 'c', 2: 'c', 3: 'a', 4: 'b'}
{1: 'c', 2: 'c', 3: 'a', 4: 'a'}
{1: 'c', 2: 'b', 3: 'c', 4: 'c'}
{1: 'c', 2: 'b', 3: 'c', 4: 'b'}
{1: 'c', 2: 'b', 3: 'c', 4: 'a'}
{1: 'c', 2: 'b', 3: 'b', 4: 'c'}
{1: 'c', 2: 'b', 3: 'b', 4: 'b'}
{1: 'c', 2: 'b', 3: 'b', 4: 'a'}
{1: 'c', 2: 'b', 3: 'a', 4: 'c'}
{1: 'c', 2: 'b', 3: 'a', 4: 'b'}
{1: 'c', 2: 'b', 3: 'a', 4: 'a'}
{1: 'c', 2: 'a', 3: 'c', 4: 'c'}
{1: 'c', 2: 'a', 3: 'c', 4: 'b'}
{1: 'c', 2: 'a', 3: 'c', 4: 'a'}
{1: 'c', 2: 'a', 3: 'b', 4: 'c'}
{1: 'c', 2: 'a', 3: 'b', 4: 'b'}
{1: 'c', 2: 'a', 3: 'b', 4: 'a'}
{1: 'c', 2: 'a', 3: 'a', 4: 'c'}
{1: 'c', 2: 'a', 3: 'a', 4: 'b'}
{1: 'c', 2: 'a', 3: 'a', 4: 'a'}
{1: 'b', 2: 'c', 3: 'c', 4: 'c'}
{1: 'b', 2: 'c', 3: 'c', 4: 'b'}
{1: 'b', 2: 'c', 3: 'c', 4: 'a'}
```

Custom

S = {1,2,3}

T = {1,2}

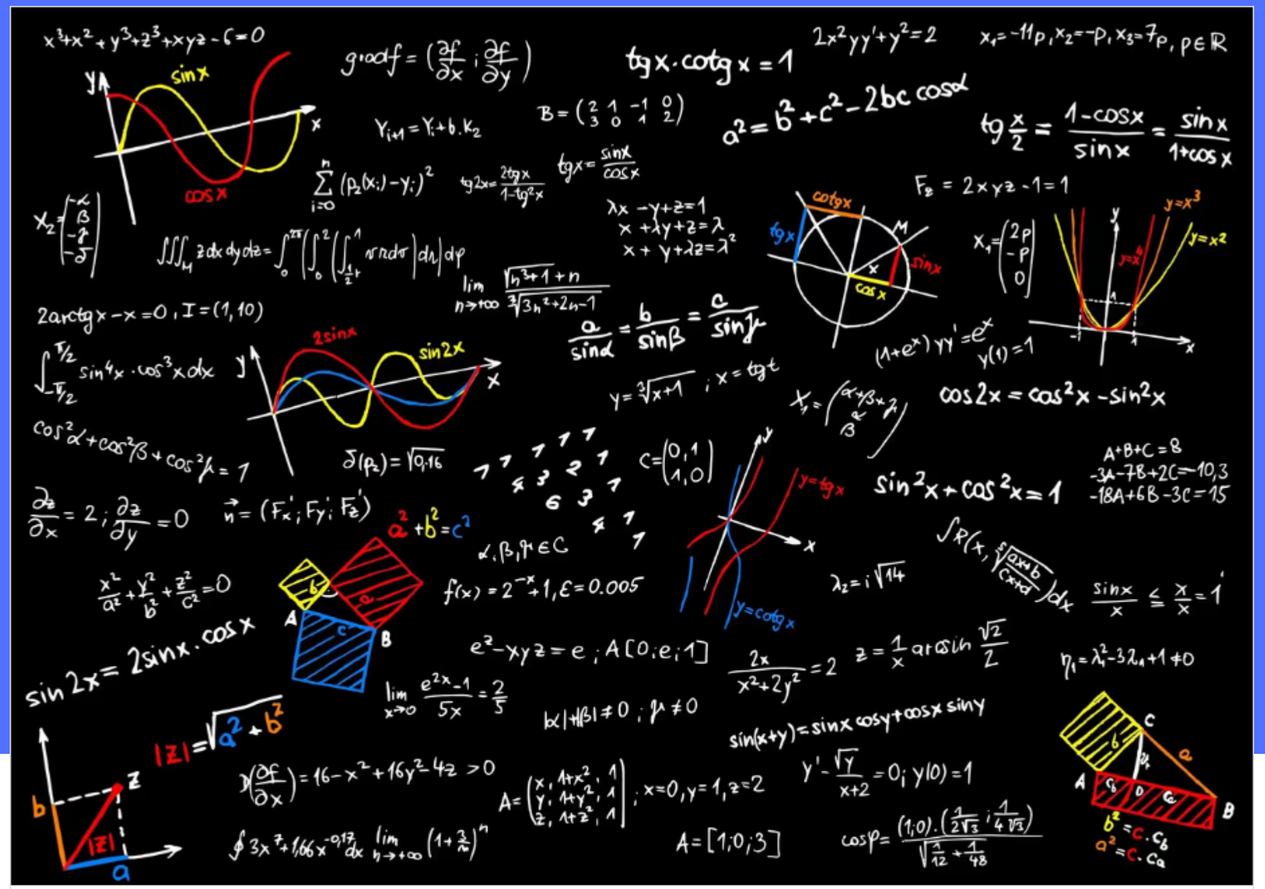
```
Try with your own sets of integers!

Enter a set S of integers separated by spaces:
1 2 3
Enter a set T of integers separated by spaces:
1 2

Number of onto functions from S to T: 8
All of the onto functions from S to T:

{1: 1, 2: 1, 3: 1}
{1: 1, 2: 1, 3: 2}
{1: 1, 2: 2, 3: 1}
{1: 1, 2: 2, 3: 2}
{1: 2, 2: 1, 3: 1}
{1: 2, 2: 1, 3: 2}
{1: 2, 2: 2, 3: 1}
{1: 2, 2: 2, 3: 2}
```

CODEBASE LINK



Hardvan/DMS-EL-Onto-Functions

Contribute to Hardvan/DMS-EL-Onto-Functions development by creating an account on GitHub.

GitHub

LINK TO THE CODE FILE

THANK YOU