**1. Candies**

Candy weighs X grams, pineapple - Y grams and apple - Z grams.

Need to write a program that will determine how many different versions of tips weighing exactly W grams can make Santa Claus.

**Input data**

The function should receive four integers X, Y, Z, and W.

**Output data**

The function should return a single integer - the number of gift options.

**Example:**

X = 10

Y = 25

Z = 15

W = 40

Result: 3

**2. Secretary Jimny**

Jimny, our secretary, went to the office late today, and it is urgently necessary to save time to have dinner, but before she needs N copies of the same document.

There are two Xerox, one of which copies the list of paper for x seconds, and the other one - for y seconds. (You may use one Xerox machine, or both at the same time. You can not only copy from the original but also use a copy.)

To help her to find out what is the minimum time it will take.

**Input data**

The function should receive three integers: N, x and y

**Output data**

The function should return a single number - the minimum time in seconds required for the preparation of N copies.

**Example1:**

N = 4

x = 1

y = 1

Result: 3

**Example2:**

N = 5

x = 1

y = 2

Result: 4

**3. CodeKiev employees**

We have the employees drink preference records in a text file (employees.json, added) one employee per line, JSON-encoded.

We want to organize the party and invite as many employees as possible. We've created the JSON files with the drink recipes (recipes.json, added) and the prices of the components (prices.json, added).

Please, write the program that will accept the party budget M and will output the names, user ids and chosen drinks for the employees that should be invited, sorted by user id (ascending).

**Input data**

// employees.json

{"id":1,"name":"Mildred Carson","drinks":["Macchiato"]}

{"id":2,"name":"Clifford Brown","drinks":["Latte"]}

{"id":3,"name":"Kellie Fletcher","drinks":["Flat White","Espresso"]}

{"id":4,"name":"Don Parsons","drinks":["Espresso"]}

{"id":5,"name":"Renee Reynolds","drinks":["Cappuccino","Macchiato"]}

{"id":6,"name":"Rudolph Bishop","drinks":["Latte","Macchiato","Flat White"]}

{"id":7,"name":"Geraldine Carpenter","drinks":["Espresso"]}

{"id":8,"name":"Hilda Jimenez","drinks":["Latte","Macchiato","Espresso"]}

{"id":9,"name":"Pauline Roberson","drinks":["Espresso"]}

{"id":10,"name":"Vanessa Barrett","drinks":["Flat White","Cappuccino","Latte"]}

// recipes.json

{

  "Cappuccino": {

    "coffee": 0.01,

    "water": 0.035,

    "milk": 0.09

  },

  "Espresso": {

    "coffee": 0.01,

    "water": 0.035

  },

  "Latte": {

    "coffee": 0.01,

    "water": 0.035,

    "milk": 0.135

  },

  "Flat White": {

    "coffee": 0.02,

    "water": 0.04,

    "milk": 0.11

  },

  "Macchiato": {

    "coffee": 0.01,

    "water": 0.035,

    "milk": 0.015

  }

}

// prices.json

{

  "coffee": 3.6,

  "water": 1,

  "milk": 1.5

}

**Output data**

The function should return names, user ids and chosen drinks for the employees that should be invited, sorted by user id (ascending).

**Example1:**

M = 0.1

Result: [{id: 4, name: "Don Parsons", drinks: ["Espresso"]}]

**Example2:**

M = 0.25

Result: [

  {id: 4, name: "Don Parsons", drinks: ["Espresso"]},

  {id: 7, name: "Geraldine Carpenter", drinks: ["Espresso"]},

  {id: 9, name: "Pauline Roberson", drinks: ["Espresso"]}

]