Technical University of Denmark

Course name	Introduction to programming and data processing
Course number	02633
Aids allowed	All Aid
Exam duration	2 hours
Weighting	All exercises have equal weight

Contents

Submission details

Two identical copies of your solutions must be handed in:

1. You must upload all your solutions on CampusNet.
2. You must upload your solutions individually on CodeJudge under *Exam*.

Note that when you hand in a solution on CodeJudge, the test example given in the assignment description will be run on your solution. If your solution passes this single test, you will appear as *Submitted*. This only means that your solution passes on this single test example. After the exam, your solutions will be evaluated on a range of different tests, to check that they works correctly in general.

Assignment A Age groups

The following table lists the names of different age groups:

Age group	Age, x
`Infant`	$x < 1$
`Toddler`	$1 \leq x < 3$
`Child`	$3 \leq x < 13$
`Teenager`	$13 \leq x < 20$
`Adult`	$20 \leq x$

Problem definition

Create a function named `computeAgeGroup` that takes the age as a numeric input and returns the age group as a string (written exactly as in the table above.)

Solution template

```
def computeAgeGroup(age):
  #insert your code
  return ageGroup
```

Input

`age`	Age in years (decimal number).

Output

`ageGroup`	Age group (string).

Example

If the age is $x = 2.9$ years, the string `Toddler` must be returned. If the age is $x = 3$ years, the string `Child` must be returned.

A

Assignment B Shirt sizes

The following table lists the sizes of mens shirts that fit for different measurements of the chest and waist.

Size	Chest	Waist
`Small`	38–42	30–35
`Medium`	40–44	32–37
`Large`	42–46	34–39
`X-Large`	44–48	36–41
`XX-Large`	46–50	38–43
`Not available`	–	–

Problem definition

Create a function named `computeShirtSize` that returns the shirt size as a string (written exactly as in the table above) such that the given chest and waist measurement are both within the given ranges. Note that the measurement ranges overlap: If the given chest and waist measurements fit more than one shirt size, the function must return the smallest size that fits. If the given chest and waist measurements do not fit any of the given shirt sizes, the string `Not available` must be returned.

Solution template

```
def computeShirtSize(chest, waist):
  #insert your code
  return shirtSize
```

Input	
`chest`	Chest measurement (decimal number).
`waist`	Waist measurement (decimal number).

Output	
`shirtSize`	Shirt size (string).

Example

Consider a chest measurement of 43.5 and a waist measurement of 34.2. The chest measurement is within the ranges for both Medium and Large, whereas the waist measurement is within the ranges for Small, Medium, and Large. Since both measurements are within the range for Medium and Large, the smallest of the two sizes is chosen and the string `Medium` must be returned.

B

Assignment C Air conditioning system

An advanced computer controlled air conditioning system can be in the different states given in the table below.

State code	Description
0	Off
1	Auto
2	Manual
3	Disabled
4	Fault

Every time the state of the system changes, the system logs the new state code as well as the time of the change as a *time stamp* measuring the number of seconds since the beginning of the log. When the system is reset, it will always make a log entry with time stamp 0 and state code 0 (Off).

Problem definition

Create a function named `acState` that returns the number of seconds the system has been in each of the five states, measured from the first logged state until the last logged state change. Input to the function is a vector of N state codes and a vector of same length with the corresponding time stamps. The function must work for vectors of any length N.

Solution template

```
def acState(state, timeStamp):
  #insert your code
  return stateTime
```

Input

`state`	State code (vector of length N).
`timeStamp`	Time stamp (vector of length N).

Output

`stateTime`	Number of seconds system has been in each of the five states (vector of length 5).

Example

Consider the following state sequence and time stamps:

State code	0	1	2	3	2	3	1
Time stamp	0	486	849	1250	2340	3560	7045

The system has been

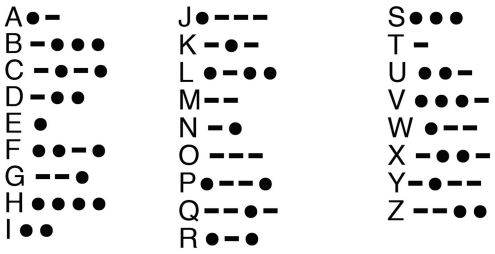
- in state 0 for 486 seconds,
- in state 1 for $849 - 486 = 363$ seconds,
- in state 2 for $1250 - 849 + 3560 - 2340 = 1621$ seconds,
- in state 3 for $2340 - 1250 + 7045 - 3560 = 4575$ seconds and,
- in state 4 for 0 seconds.

Thus the function must return the vector [486, 363, 1621, 4575, 0].

C

Assignment D Morse code

Morse code is a method of transmitting messages using a series of short and long signals called *dots* and *dashes.* The Morse alphabet is shown below:



In this assignment, we represent a Morse code message as a string, where a period represents a dot and a hyphen represents a dash. Each letter is separated by a space and each word is separated by two spaces.

Problem definition

Create a function that converts a string written in Morse code into a string written in the usual alphabet with letters A–Z (and space between words). The returned text must be in capital letters.

Solution template

```
def morseToText(morseCode):
  #insert your code
  return text
```

Input	
`morseCode`	Message written in the Morse alphabet (string containing only dot, dash, and space characters).

Output	
`text`	Message written in the normal alphabet (string containing only letters A-Z and space).

Example

Consider the following input string:

```
-- --- .-. ... .  -.-. --- -.. .
```

To clearly show the space characters in the string, we can mark them visually by a ␣ symbol:

```
--␣---␣.-.␣...␣.␣␣-.-.␣---␣-..␣.
```

For this input string, the output must be the string

```
MORSE CODE
```

D

Assignment E The D'Hondt method

The D'Hondt method is an algorithm that is used to allocate seats to parties in elections, based on their number of votes. In general, the number of seats for each party should be proportional to the number of votes received; however, achieving exact proportionality is often not possible. In the D'Hondt method, the seats are assigned to parties one by one, to the party has the highest quotient, q, given by

$$q = \frac{V}{s+1}, \tag{1}$$

where V is the number of votes the party reveived and s is the number of seats assigned to the party so far. If two quotients of two parties are identical, the seat is assigned to the party with the highest number of votes, and we assume that no two parties have an identical number of votes.

Problem definition

Create a function which takes a vector of number of votes for N parties as input as well as the number of seats, and returns the seat assignment computed using the D'Hondt method as described above.

Solution template

```
def dhondt(votes, seats):
  #insert your code
  return seatsAllocated
```

Input	
`votes`	Number of votes for each of N parties (vector of length N).
`seats`	Number of seats to assign (positive whole number).

Output	
`seatsAllocated`	Number of seats assigned to each of the N parties (vector of length N).

Example

Consider the situation where five parties have received the following number of votes and there are seven seats to be assigned. The table shows the quotients for each of the parties as the seats are assigned one by one. The highest quotient, which is marked in bold, gives one seat to the respective party.

	Party A	Party B	Party C	Party D	Party E
Votes	340 000	280 000	160 000	60 000	15 000
Seat 1	**340 000**	280 000	160 000	60 000	15 000
Seat 2	170 000	**280 000**	160 000	60 000	15 000
Seat 3	**170 000**	140 000	160 000	60 000	15 000
Seat 4	113 333	140 000	**160 000**	60 000	15 000
Seat 5	113 333	**140 000**	80 000	60 000	15 000
Seat 6	**113 333**	93 333	80 000	60 000	15 000
Seat 7	85 000	**93 333**	80 000	60 000	15 000
Seats	3	3	1	0	0

At first, the quotient for each party is identical to the number of votes. The first seat is assigned to Party A, and the quotient for Party A is updated to $\frac{340\,000}{2} = 170\,000$. The second seat is assigned to Party B, which now has the highest quotient, and the quotient for Party B is updated to $\frac{280\,000}{2} = 140\,000$. The third seat is assigned to Party A, which now again has the highest quotient, and the quotient of Party A is updated to $\frac{340\,000}{3} = 113\,333$. The assignment process continues until all seats have been assigned. The output of the function in this example must be the vector of the number of seats for each party,

$$[3,\ 3,\ 1,\ 0,\ 0]$$

E