

Introductory course in programming (DAT425)

5 January 2021

Examiner: Krasimir Angelov

Note: Solutions must be uploaded to Canvas as a single Python file. It is not necessary to comment on your code, it is more important that the code is clean and easy to read.

All help materials are allowed, but everyone is expected to work alone. Cooperation is not allowed and cheating may lead to suspension from the university.

10 points correspond to grade 3, 17 points correspond to grade 4, and 24 points correspond to grade 5.

Question 1

In the United States, one is assumed to be an adult, when the age of 18 is reached. However, 21 years are required to purchase alcohol. 30 years to serve as a Senator, and 35 to serve as president.

Write the function `legal_status(age)` which returns one of the strings `'minor'`, `'adult'`, `'alcohol'`, `'senator'`, or `'president'`, depending on the parameter `age`.

Add the function `main()` which asks the user for his/her age and prints out his legal status. The main function should take care of the input/output while the pure computations should be

left to `legal_status`. The `legal_status` on the other hand should not do any input/output.

This is an example dialog with the program:

```
>>> main()
```

```
What is your age: 20
```

```
Your legal status is: adult
```

(7 points)

Question 2

Implement the function:

```
duplicates(xs)
```

Which takes a list of elements and returns a new list which only contains those elements which appear more than once in the original list. One way to solve the problem is to count how many times each element appears, and then to preserve only those for which the count is greater than one.

Here are some examples:

```
>>> duplicates([1,2,3,2,1,1,5])
```

```
[1,2]
```

```
>>> duplicates(['a','b','b','a'])
```

```
['b']
```

```
>>> duplicates([True,False])
```

```
[]
```

```
>>> duplicates([])
```

```
[]
```

(10 points)

Question 3

Implement a class called Doorlock which models a door lock which can be opened with a pin code. At any time the door lock is either opened or closed and it is set up with a specific pin code.

The class should have the following methods where all methods except the `__init__` method must return True or False, depending on whether the operation was successful:

- The init method should set up the door lock in opened state and the pin code should be set up according to the parameter:

```
def __init__(self, pin): ...
```

- The lock method:

```
def lock(self): ...
```

is successful only if the door lock is currently open. After successful execution of the method the lock must close.

- The unlock method:

```
def unlock(self, pin): ...
```

is successful only if the current pin code matches the code in the argument, and only if the door lock is currently closed. After successful execution of the method the door lock is open.

-

- The `set_pin` method:

```
def set_pin(self, pin, new_pin): ...
```

is successful only if the door lock is currently open and its current pin code matches the argument “pin”. After successful execution of the method the pin code is changed to the argument “new_pin”. The door lock remains open.

(13 points)