MATEMATISKA INSTITUTIONEN
STOCKHOLMS UNIVERSITET
Avd. Beräkningsmatematik
Kursledare: Anders Mörtberg
Examinator: Lars Arvestad

Tentamensskrivning i
DA2004 Programmeringsteknik för matematiker
7.5 hp
2022-03-14

- Tentan har flervalsfrågor där minst ett svarsalternativ är korrekt. Om man svarar fel eller inte har exakt antal rätta alternativ får man noll poäng på frågan.
- **Skriv tydligt.** Svårlästa svar riskerar 0 poäng.
- Skriv bara på en sida av varje papper!
- Man måste bli godkänd på del A (5 rätt på 10 frågor) för att del B ska rättas.
- **Hjälpmedel:** Ett A4 med så mycket information du vill. Du får skriva på båda sidorna.
- **Betygsgränser:** E: 10, D: 12, C: 14, B: 16, A: 18, av maximala 20.

Del A: flervalsfrågor

Var snäll och samla svaren på del A på ett svarspapper. Varje fråga på del A är värd 1 poäng.

1. Consider the code snippet to the right. What is the value of `s` after running the code?

A. 3
B. 8
C. 11
D. Nothing, an exception is raised because dictionaries cannot contain lists.
E. Nothing, an exception is raised because you cannot calculate the length of a dictionary.

```
d = { 2 : "hej",
      3 : [1,2,3],
      4 : { 5 : "hej" , 6 : "du" } }

s = 0

for x in d:
    s += len(d[x])
```

2. What is printed if we run the code snippet to the right?

A. Nothing, because we have not called `f` with enough arguments.
B. 25
C. 7
D. Nothing, `f` goes into an infinite loop because we have written `y = x + y`.
E. 6

```
x = 10

def f(x,y,z=3):
    y = x + y
    return (x + y + z)

print(f(1,2))
```

3. Consider the list comprehension `[ x * 2 for x in range(0,100) if x % 2 == 0 ]`. Which of the code snippet below produces the same list?

A. `list(map(lambda x: x * 2, reduce(lambda x: x % 2 == 0,range(0,10))))`
B. `list(filter(lambda x: x % 2 == 0, map(lambda x: x * 2,range(0,10))))`
C. `list(map(lambda x: x * 2, range(0,10)))`
D. `list(map(lambda x: x * 2, filter(lambda x: x % 2 == 0,range(0,10))))`
E. None of the alternatives above. You need to use a **`for`** or **`while`** loop to produce the list.

4. Which of the following assignments make `x or (not y and z)` evaluate to `True`?

A. `x = True`, `y = False`, and `z = True`
B. `x = False`, `y = False`, and `z = True`
C. `x = True`, `y = True`, and `z = True`
D. `x = False`, `y = False`, and `z = False`
E. `x = False`, `y = True`, and `z = True`

5. Consider the code snippet to the right. What is the result of evaluating `f([9,2,1,3])`?

A. 15
B. 6
C. 10
D. 4
E. 3

```
def f(mylist):

  if mylist == []:
      return 0
  else:
      return len(mylist) + f(mylist[1:])
```

6. Which of the following statements are true in Python?

A. The term *higher order functions* refers to functions that can handle arbitrarily large numbers.
B. Subclasses can have only one superclass.
C. Strings are mutable.
D. Lists are mutable.
E. **`range`** returns a "**`range`**-object", not a list.

7. What keyword is used to create an exception in Python?

A. `try`
B. `except`
C. `throw`
D. `raise`
E. `exception`

8. Consider the code snippet to the right. What happens if we run `divide(2,0)`?

A. An `AssertionError` is raised.
B. The number `2` is returned.
C. A `ZeroDivisionError` is raised.
D. The number `0` is returned.
E. First `AssertionError: assert y != 0` is printed, then a `ZeroDivisionError` is raised.

```
def divide(x,y):
    assert y != 0
    return x / y
```

9. What happens if we run the code snippet to the right?

A. Nothing, we go immediately into an infinite loop.
B. The user inputs numbers until they enter `42`, then the loop ends and `Congratulations!` is printed.
C. Nothing, as one cannot write a **`while`** loop with a variable but must have a boolean test.
D. The user inputs numbers, but the loop never ends because `x` is not converted to an integer before it is compared with `42`.
E. The user can enter numbers, and `Congratulations!` is printed each time they enter `42`, but the loop never ends because there is no **`break`**.

```
b = True

while b:
    x = input("Write a number: ")

    if x == 42:
        print("Congratulations!")
        b = False
```

10. Consider the code snippet to the right. What is the value of `out` after running this code?

A. `Pyyttthhhhooooonnnnnn`

B. `Python`

C. `nnnnnnooooohhhhtttyyP`

D. `nnnnnoooohhhtty`

E. `ytthhhoooonnnnn`

```
s = "Python"

out = ""

for i in range(0,len(s)):
    out += i * s[i]
```

Del B: koduppgifter

Var snäll använd ett papper till varje uppgift i del B.

11. Below is an attempt at writing a program to calculate all numbers smaller than `v` that divide `v`:

```
v = 42
divs = []

for x in range(0,v):
    if v % x == 0:
        divs.append(x)
```

This code however contains a bug and will raise an exception if we try to run it. Your task is to modify the code so that is works like it should, and improve it by creating a function `divisors(v)` which takes in `v` as an argument instead of a global variable. (1p)

Sample execution:

```
[In: ] print(divisors(42))
[Out:] [1, 2, 3, 6, 7, 14, 21]
[In: ] print(divisors(62))
[Out:] [1, 2, 31]
```

12. Write a function `exponents(x,mylist)` using **`map`** and **`lambda`** that takes in a number `x` and a list of numbers `mylist` and then returns a list containing `x` raised to the power of each number in the list.

Sample execution:

```
[In: ] print(exponents(2,[1,2,3]))
[Out:] [2, 4, 8]
[In: ] print(exponents(3,[5,2,1,3]))
[Out:] [243, 9, 3, 27]
```

Obs: to receive credit, `exponents` must be written wtih **`map`** and **`lambda`**.

13. Write a function `word_length(f)` which takes in a filename `f` and prints out the length of every word in the file with a space between each word length and line breaks in the same places as in the original file.

Sample execution: given a file `indata.txt` that contains

```
Hej hej,
lycka till på tentan!
```

`word_length('indata.txt')` should print:

```
3 4
5 4 2 7
```

Obs: special characters such as `','` and `'!'` should be treated as letters.

14. ASCII is a standard for representing letters as numbers. For example, "a" is represented as 97, "b" is represented as 98, etc., all the way up to "z" which is represented as 122. Python has a function **`chr`** that returns a letter given a number according to the ASCII standard, so **`chr`**`(97)` is `'a'`, **`chr`**`(98)` is `'b'`, etc.

Write a code snippet that uses **`chr`** to create a dictionary `letters` from lowercase letters to their position in the alphabet, counted starting from `1` (so the key `'a'` should have the value `1`, the key `'b'` should have the value `2`, etc.). (1p)

Obs: to receive credit, you must not hardcode the dictionary. It should be created by using the **`chr`** function in an appropriate way, for example using a loop or dict-comprehension.

15. Write a function `sum_string(s,d)` that takes in a string `s` and a dictionary `d` from letters to numbers, and returns the sum of the numbers that the letters in `s` correspond to in `d`. (1p)

Sample execution with the dictionary `letters` as defined in task 14 above:

```
[In: ] print(sum_string("hej",letters))
[Out:] 23
```

We get this result because "h" is `8`, "e" is `5` and "j" is `10`.

16. Write a recursive function `even_odd_sum(mylist)` that, given a list `mylist` of integers, calculates the sum of the numbers in such a way that even numbers are added whereas odd numbers are subtracted.

Sample execution:

```
[In: ] print(even_odd_sum([1,2,3]))
[Out:] -2
[In: ] print(even_odd_sum([2,4,6,8]))
[Out:] 20
[In: ] print(even_odd_sum([1,3,6,3]))
[Out:] -1
```

Obs: to receive credit, your function must be recursive.

17. In the *Lord of the Rings* books, orcs are a type of monster. Write a class `Orc` so that when one calls its constructor, the *instance attributes* `hp` and `strength` are set with values provided by the caller. These attributes represent the orcs life energy and strength. Write another method `beats` that, given another orc, checks who is the strongest and returns `True` if the orc calling the method is at least as strong or stronger than the other orc and `False` otherwise. (1p)

Sample execution:

```
[In: ] orc1 = Orc(10,20)          # Orc with 10 HP and 20 strength
[In: ] orc2 = Orc(8,10)           # Orc with 8 HP and 10 strength
[In: ] print(orc1.beats(orc2))    # orc1 beats orc2 as it has higher strength
[Out:] True
```

18. A special type of orc is Uruk-hai. These obey the wizard Saruman and are extra strong and powerful. Define a class `UrukHai` that inherits from `Orc` and has the *class attribute* `master` set to `'Saruman'` and a constructor that uses the constructor of the superclass `Orc`, but doubles both the `hp` and `strength` that that the Uruk-hai is created with.

Sample execution:

```
[In: ] urukhai = UrukHai(15,15)      # Uruk-hai with 2*15 HP and 2*15 strength
[In: ] print(urukhai.beats(orc1))    # The Uruk-hai beats the stronger of the two orcs
[Out:] True
[In: ] print(orc2.beats(urukhai))    # so the weaker orc also doesn't beat the Uruk-hai
[Out:] False
[In: ] print(urukhai.master)         # The master of all Uruk-hai is Saruman
[Out:] Saruman
```

19. Write a program that asks the user what the product of two randomly chosen numbers between 0 and 10 is. If the user enters the correct product, the program should congratulate them and begin again with two new numbers. If the user instead enters an incorrect product, the program should inform the user and print out the correct program, then begin again. If the user doesn't write anything (i.e. enters the empty string `''`), the program should end. (1p)

Furthermore, if the user enters something that is not a number or the empty string, the program should not crash; any error should be handled in an appropriate way using **`try`**-**`except`**. (1p)

Sample execution:

```
What is 8 * 9? 78
Incorrect! The correct answer is 72
What is 9 * 7? 63
Correct!
What is 2 * 5? apabepa
Input is not a number! The correct answer is 10
What is 1 * 0? 0
Correct!
What is 7 * 7?
Goodbye! The correct answer was 49
```

In the example above, the user has entered `78`, `63`, `apabepa`, `0`, and finally nothing, at which point the program exits.

You can use the function `randint` from the `random` library in this task. It works in such a way that `randint(0,10)` returns a random number between `0` and `10`.