MATEMATISKA INSTITUTIONEN
STOCKHOLMS UNIVERSITET
Avd. Beräkningsmatematik
Lecturer: Kristoffer Sahlin
Examinator: Kristoffer Sahlin

Exam in
DA7066 Programming techniques for Life Sciences
7.5 hp
2024-04-03

- If a multiple choice question has several correct answers, you must give all correct answers to receive credit.
- **Write clearly.** Answers that are difficult to read may receive 0 points.
- You must pass part A (4 correct out of 8 questions) to have your part B graded.
- Write only on one side of each paper!
- No **`import`** are allowed in your answers in part B (neither Python's standard modules nor external libraries) unless it is explicitly mentioned in the assignment. Built-in functions like **`len`**, **`range`**, **`print`**, and **`sum`** are allowed.
- **Aids:** An A4 with as much information as you want. You can write on both sides.
- **Grade thresholds:** E: 10, D: 12, C: 14, B: 16, A: 18, of maximum 20.

Part A: Multiple choice

Please collect your answers to part A on a single piece of paper.

1. What is printed by the code to the right?

A. -4
B. -3
C. 15
D. 33
E. An exception is raised.

```
y = 15
for x in [4, 10, 4]:
    y -= x
print(y)
```

2. Which list does the function `fcn` return?

A. `[]`
B. `[1, 2, 3, 4, 5]`
C. `[2, 4, 6, 8, 10]`
D. `[1, 4, 9, 16, 25]`
E. `[2, 2, 2, 2, 2]`

```
def fcn():
    res = []
    for i in range(5):
        print(i**2)
    return res
```

3. Which of these are operators for logical expressions in Python?

A. `and`
B. `not`
C. `or`
D. `maybe`
E. `probably`

4. What will be the value of the variable `text` after running the code to the right?

A. `top`
B. `toptoptop`
C. `ttt`
D. `ppp`
E. `pot`

```
text = ""
msg= "top"
i = len(msg)
for j in range(len(msg)):
    while j < i:
        text += msg[j]
        i -= 1
```

5. Which the following statements are true for Unix (assuming the bash interpreter as used in the course compendium)?

A. `cd` is used to change current working directory.
B. `rm` **`file`**`.txt` creates a new file **`file`**`.txt`
C. `mkdir` is used to remove files and directories.
D. `wc` can count lines, characters, and words in files.
E. `cp old.txt new.txt` will create a copy, `new.txt`, of the file `old.txt`.

6. What is the result of the code on the right?

A. `'Bird'`
B. `'Door'`
C. `'Sky'`
D. `'Lake'`
E. An error message

```
d = { 1 : {'a': 'Bird', 'b': 'Door'},
      2 : {'b': 'Sky', 'a': 'Lake'}}
print(d[0]['b'])
```

7. What is printed by the code on the right?

A. 1
B. 2
C. 3
D. 4
E. 8

```
a = 1
b = 2
def some_fcn(c, d):
    x = a * b + c * d
    return x

def another_fcn(c, d):
    return some_fcn(a, b) + some_fcn
    (c, d)

print(another_fcn(a, b))
```

8. What will be the contents of `d` after the code to the right is run?

A. `{'D': 6, 'A': 6, '7': 6, '0': 6, '6': 6 }`
B. `{'D': 0, 'A': 1, '7': 2, '0': 3, '6': 9}`
C. `{'D': 0, 'A': 1, '7': 2, '0': 3, '6': 5}`
D. `{'D': 0, 'A': 1, '7': 2, '0': 3, '6': 4}`
E. `{0: 'D', 1: 'A', 2: '7', 3: '0', 4: '6', 5: '6'}`

```
s = 'DA7066'
d = {}
for x in s:
  d[x] = s.index(x)
```

Part B: Coding

Please use a separate piece of paper (or several) for each question in part B. Multipart questions such as 9A and 9B can be written on the same piece of paper.

9.

A. Write a function `my_special_sum` that sums a list of integers such that even integers are added to the sum and odd integers are subtracted from the sum. For example, `[1,2,3]` becomes $-1 + 2 - 3 = -2$. (2p)

Example use:

```
[In: ] print(my_special_sum([6,2,4]))
[Out:] 12
[In: ] print(my_special_sum([2,3,4]))
[Out:] 3
[In: ] print(my_special_sum([1,1,1]))
[Out:] -3
```

B. Make `my_special_sum` able to handle lists that contain elements which are not integers (for example **str**, **float**, **bool**) by printing `'invalid datatype at list at index: X'` for the items where this happens, where `X` is the position (i.e., index) of the variable in the list. (1p)

Tip: You can use the built-in function **type** that returns the type of the variable.

Example usage:

```
[In: ] print(my_special_sum([7.0,6,2,4]))
[Out:] invalid datatype at list at index: 0
[Out:] 12
[In: ] print(my_special_sum([1,'1',1,1, False]))
[Out:] invalid datatype at list at index: 1
[Out:] invalid datatype at list at index: 3
[Out:] -3
```

10. The code below is intended to reverse the string passed as an argument to `rev`, see the examples on the right. However, there are at least two errors. Explain what they are. (2p)

```
i = 1
def rev(s):
    s_rev = []
    while i <= len(s):
        s_rev += s[-i]
        i = i + 1
    return s_rev
print(rev('hubba'))
```

```
[In: ] rev('hubba')
[Out:] 'abbuh'
[In: ] rev('foo')
[Out:] 'oof'
```

11. Write a correct version (i.e., works as in the two examples) of the function `rev` in question 10 that uses a `for` loop instead of a while loop and raises a `ValueError` if an empty string is passed. (2p)

12. Write a function that constructs a dictionary by reading in keys and values from the user. If the user enters a key that has already been added to the dictionary, the function should print that the key has already been added, and give the user a new chance to enter a key. If the user enters `'quit'` as the key, the function should return the dictionary. You don't need to do any error handling. (2p)

Example usage:

```
[In: ] d = construct_dict()
[Out:] Provide key: dna
       Provide value: ACGT
       Provide key: dna
       The key is already in the dictionary.
```

```
        Provide key: hi
        Provide value: 99
        Provide key: quit
[In: ] print(d)
       {'dna': 'ACGT', 'hi' : '99'}
```

13. Write a function `common_max(list1,list2)` that takes two lists with integers as arguments and returns the largest common integer appearing in both lists. If there is no common integer, `0` should be returned. (3p)

Note: You can use the built-in function **`max`** that returns the maximum value in a list.

Example use:

```
[In: ] print(common_max([6,1,2,4],[2,3,5]))
[Out:] 2
[In: ] print(common_max([6,1,2,3,4],[4,2,3,6,7]))
[Out:] 6
[In: ] print(common_max([6,1,4],[2,3,5]))
[Out:] 0
```