

Python for Rookies Example Examination:

Answer Book

Question 1.

a) **Class variable:** interest_rate, **instance variables:** name, number, balance.

[5 marks]

b)

```
def deposit(self, amt):
    self.balance += amt
    return
def withdraw(self, amt):
    if self.balance - amt > 0:
        self.balance -= amt
    return
def add_interest(self):
    i = BankAccount.interest_rate *
        self.balance
    self.balance += i
```

[10 marks]

c)

```
class StudentAccount(BankAccount):
    overdraft_lmt = 1000
    def __init__(self, name,
                 number, balance):
        BankAccount.__init__(self, name,
                              number, balance)
        return
    def withdraw(self, amt):
        if self.balance - amt > overdraft_lmt:
            self.balance -= amt
        return
```

[10 marks]

Question 2.

```
class Palindrome(object): # [2 marks]
    def __init__(self):
        pass
    def reverse(self, s): # [8 marks]
        rev = ""
        for i in range(len(s)):
            rev += s[-i-1]
        return rev
    def isPalindrome(self, s): [5 marks]
        if s == self.reverse(s):
            return True
        else:
            return False

def main(): # [10 marks]
    p = Palindrome()
    cases = ['dad', 'peep', 'deed',
            'dude', 'task', 'foobar']
    for case in cases:
        print 'reverse(', case, '):',
            str(p.reverse(case))
    for case in cases:
        print 'isPalindrome(', case, '):',
            str(p.isPalindrome(case))
```

Award more marks for good coding practices, e.g. testing boundary conditions, storing test cases in a linear data structure (ease of extension), and so on.

Question 3.

a)

```
def sum(n): # iterative
    total = 0
    for i in range(n):
        total += i
    return total
```

[4 marks]

b)

```
def sum(nums): # iterative
    total = 0
    for i in nums:
        total += 1
    return total
```

[5 marks]

c)

```
i=1
while(i<20):
    print "i = ", i
    i += 1
```

Be generous about off-by-one errors.

[8 marks]

d)

```
for i in range(20, 0, -1):
    print "i = ", i
```

Be generous about off-by-one errors.

[8 marks]

Question 4.

a) Anything equivalent to:

```
for i in range(360):  
    forward(1)  
    left(1)
```

[5 marks]

b) Any algorithm is acceptable so long as it covers the entire frame.

[5 marks]

c) Consideration should be given to how the turtle can tell if it has moved off the track. One strategy is to move the turtle forward a pixel or so, then rotate left and right by increasing amounts until one of the sensors detects the track.

[15 marks]

Question 5.

a) A recursive function calls itself.

[2 marks]

b)

```
def sum(arg):
    if arg == []:
        return 0
    else: return arg[0] + sum(arg[1:])
```

Award two marks for the base case and three for the recursive case. Do not penalise students for not using slicing.

[5 marks]

c) `foobar` reverses a list. e.g. `foobar([1, 2, 3])` returns `[3, 2, 1]`.

Award half marks for stating what the method does, two for a suitable argument and two for a correct return value.

[8 marks]

d)

- `iso(3) -> True`
- `iso(2) -> False`
- `ise(3) -> False`
- `ise(2) -> True`
- `iso` returns `True` if the argument is odd and `False` otherwise. `ise` returns `True` if the argument is even and `False` otherwise.

Award 2 marks each. Candidates should show their workings.

[10 marks]