IT1007 Midterm Quiz

Question 1 [10 marks]: Computing the Natural Number *e*

The mathematical constant e is the unique number whose natural logarithm is equal to one. And you can compute the number with the following sum the infinite series,

$$e = \sum_{n=0}^{\infty} \frac{1}{n!} = \frac{1}{1} + \frac{1}{1} + \frac{1}{1 \times 2} + \frac{1}{1 \times 2 \times 3} + \frac{1}{1 \times 2 \times 3 \times 4} + \cdots$$

Of course, we cannot compute the perfect e until n equals to infinity. However, we can compute n to a certain steps. E.g.

$$\sum_{n=0}^{1} \frac{1}{n!} = 2$$

$$\sum_{n=0}^{2} \frac{1}{n!} = 2.5$$

$$\sum_{n=0}^{100} \frac{1}{n!} = 2.7182818284590455$$

Write a function '`compute_e(n)`' such that it compute e from the terms 0 to n. You can assume that there is a function **`factorial(k)`** that compute the factorial of k. Here are some sample outputs:

```
>>> compute_e(0)
1.0
>>> compute_e(1)
2.0
>>> compute_e(2)
2.5
>>> compute_e(100)
2.7182818284590455
>>> compute_e(5000)
2.7182818284590455
```

Answers:

```
def compute_e(n):
```

Question 2 [4 marks]

Each row of the table is a separate program/file. What is the output of each of them? If the code produces errors or runs into infinite loops, please state 'error' or 'infinite loop' respectively.

Code	Output

`l = [1,2]`
`a = 1`
 	
`def double(x):`
`    x += x`
 	
`double(l)`
`double(a)`
 	
`print(l)`
`print(a)`	

`b = 128`
 	
`def half(x):`
`    return x / 2`
 	
`print(half(half(half(b))))`	