Chu Thành Bách

**1. What is the lambda function in Python? Why does it exist in Python?**

- Lambda is a function in Python that can take one or many arguments, but has only one expression. Lambda syntax:

**lambda** *arguments* : *expression*

- The power of lambda is better shown when you use them as an anonymous function inside another function.

**def** myfunc(n):

**return** **lambda** a : a \* n

mytripler = myfunc(3)

**print**(mytripler(11))

**Use lambda functions when an anonymous function is required for a short period of time.**

**2. What is pass in Python?**

The pass statement is used as a placeholder for future code.

When the pass statement is executed, nothing happens, but you avoid getting an error when empty code is not allowed.

**3. What is \*args, \*\*kwargs in function definition?**

- \*args used when you don’t know the number of argument but know what to do with the first and extra arguments (non-key worded)

- \*kwargs used when you don’t know the number of argument but know what to do with the first and extra arguments (key worded)

**4. What is docstring in Python? How to write them? Are they required?**

- The first string after the function is called the Document string. This is used to describe the functionality of the function.

- The first line after the function, write: # + how you describe the function.

- The use of docstring in functions is optional but it is helpful when other people want to know more about your function (what it for, how to use,…).

**5. What are the built-in data types that Python provides? Which of them are mutable, which are immutable?**

- The built-in data types for Python are:

+ dict: indexed (has key), mutable, unordered, only unique value

+list: indexed, mutable, ordered

+set: unindex, immutable, unordered, only unique value

+tuple: indexed, immutable, ordered

**6. What is the difference between list and tuple types in Python?**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Syntax | Mutability | Use |
| List | [a1,a2,…] | mutable | Store a list of data that can be changed |
| Tuple | (a1,a2,…) | immutable | Store data that can not be changed |

**7, What keywords can be used in conjunction with the *for* keyword?**

- while, break, continue, if, elif, else (Câu này chưa thật sự hiểu lắm)

**8, What's the difference between *globals(), locals(),* and *vars()*?**

- **globals()** returns the dictionary of the current global symbol table. Global symbol table stores all information related to the global scope of the program, which contains all functions, variables that are not associated with any class or function.

- **locals()** returns the dictionary of the current local symbol table. Local symbol table stores all information related to the local scope of the program, which contains all functions, variables that are within a class or function.

- **vars(***object***)** returns the \_\_dict\_\_ attribute of the given object.

**9, Is it possible to have a negative index in iterative types in Python?**

No

**10. What is the \_\_init\_\_.py module? What it's for?**

It is a module inside a package that initialization the nessesary base values and other module when you import the package.

**11. How can I swap values of variables in Python? Please give an example**

- Use temporary variable to swap x,y:

temp=x

x=y

y=temp

- Without temporary variable:

x, y = y, x

**12. How do I view object methods? Please give an example**

**13. What is a module in python? What is a package? What is the difference between packages and modules in Python? Please give an example module and package**

Module is a .py file in the same folder of the main .py file that can be imported. Package is a folder that has modules, sub package and \_\_init\_\_.py which can be imported as you wish.

**14. What is the \_\_init\_\_ function used for?**

It helps to initiate base values in a class

**15. Explain how to make a Python script executable on Unix?**

**16. What is the output of -12 % 10 and -12 // 10.**

**17. Why shouldn't you make the default arguments an empty list?**

**18. What is the yield keyword used for in Python?**

**19. What is an iterator in Python? Can you write an example?**

**20. What is the difference between \_\_iter\_\_ and \_\_next\_\_?**

**21. What is unittest module in Python? How to write tests in Python?**

**22. What are metaclasses in Python?**

**JUNIOR AND MIDDLE CODE INVOLVING**

1. **Write a function that produces the Fibonacci sequence.**
2. **How to translate a string containing a binary code (1 and 0) into a number (integer)? Write a function to do this.**

def binToInt(a):

    return int(a,2)

1. **How to check that tuple A contains all elements of tuple B. Do both tuples contain unique values? Write a function to do this.**

def comTup(a,b):

    a1=list(set(a))

    b1=list(set(b))

    sum=0

    for i in b1:

        if a1.count(i)==1:

            sum=sum+1

    if sum==len(b1):

        print('A has all B values')

    else:

        print('''A doesn't have all B values''')

    if len(set(a))<len(a) or len(set(b))<len(b):

        print('Both not unique')

    else:

        print('Both unique')

1. **How to convert a string to a number that consists of letters ASCII code. Example: 'abcd' -> 979899100. Write a function to do this.**

def strToASCII(a):

    b=list(a)

    c=[]

    sum=0

    for i in b:

        c.append(str(ord(i)))

    return (''.join(c))

1. **How to remove empty lines from a list of lines (with a length of 0). Write a function to do this.**

def delEmpty(a):

    b=[]

    for i in range(0,len(a)):

        if len(a[i])!=0:

            b.append(a[i])

    return b

1. **Write a function that returns a string of numbers from 0 to 100, "0123456789101112...".**

def str100():

    a=[]

    for i in range(0,101):

        a.append(str(i))

    return ''.join(a)

1. **Write a function that makes a list with unique items from a list with duplicate items. Example: [1, 1, 2, 3, 3] -> [1, 2, 3]**

def lstUnique(a):

    b=list(set(a))

    return b

1. **Write a program that prints the numbers from 1 to 20. But for multiples of three print “Fizz” instead of the number and for the multiples of five print “Buzz”. For numbers that are multiples of both three and five print “FizzBuzz”.**

def pr():

    a=[]

    for i in range(1,21):

        if i%3==0 and i%5!=0:

            a.append('Fizz')

        elif i%5==0 and i%3!=0:

            a.append('Buzz')

        elif i%15==0:

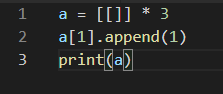
            a.append('FizzBuzz')

        else:

            a.append(str(i))

    return ''.join(a)

1. **What will be the output of the following code? Please explain why?**



1. **We have the following code with the unknown function f(). In f(), we do not want to use a return, instead, we may want to use a generator.**

https://lh3.googleusercontent.com/YmqTKJss18ChbAXYrwwguVt_UCeS7mqBUS7XFWwjBmCtj4HGjbu84MANxsWXUWuvyr-Lbh4CSCM0xqpIzxjuqMHBk9j4siBw7N4uRUZ5DtahHXkeZjKog4sXOuK6R63hjigOZF5YrcNYBM2E3w=s800

The output looks like this:

https://lh6.googleusercontent.com/bixcWurxyY3X-BbWYKkJB4PxBp5oNQ0wb_CO-1pT1sOFsQ8S_rsF4bCvDflzezzVD2lapfvw0zVFk9fj9EKDprbER0SHkwNJfUeqvykuUnRYdaOX6ofqZxYVHez0BQYO5y0jw_X_iQryYjg8RQ=s800

Write a function f() so that we can have the output above.