**西安电子科技大学**

**安全前沿讨论班（I） 课程实验报告**

**实验名称 Python入门实践案例**

学 院 网络与信息安全学院 班 级 1618019

姓 名 曹寅峰 学 号16020610025

实验日期 年 月 日

|  |
| --- |
| 指导教师评语：  指导教师：  年 月 日 |
| 1. 实验目的   练习，回顾python的基本语法，控制流等，并在最后实现一个简单的拼写检查器   1. 实验环境   Python3.5 ,Jupyter notebook   1. 实验基本原理及步骤  BMI计算 根据题意，通过计算BMI后用if判断BMI范围即可  def computeBMI(height,weight):  """compute BMI value based on height and weight"""  res=weight/(height\*height)  return res  def internationalBMI(BMI):  """return international BMI index, the input parameter is BMI value"""  res=''  if BMI>=30:  res='肥胖'  elif BMI>=25:  res='偏胖'  elif BMI>18.5:  res='正常'  else:  res='偏瘦'  return res  def domesticBMI(BMI):  """return domestic BMI index, the input parameter is BMI value"""  res=''  if BMI>=38:  res='肥胖'  elif BMI>=24:  res='偏胖'  elif BMI>18.5:  res='正常'  else:  res='偏瘦'  return res 字符串输出 题中要求保留一段英文中首字母为h-z的单词  **总体思路为：**   1. **将输入字符串分割成单词列表** 2. **利用ord函数判断每个单词的首字母是否在范围内，如果在范围内，添加到一个新列表中，防止引用出错**   oristr=input('input your string')  strlist=list(oristr.lower().split(' '))  reslist=[]  3print(strlist)  for word in strlist:  if ord(word[0])<ord('h') or ord(word[0])>ord('z'):    continue  else:  reslist.append(word)  for word in reslist:  print(word.upper()) 字符串操作 题中要求输入一个列表和一个字符串，对列表进行操作，字符串如果在原来列表中存在则弹出一个，不存在则加入，空输入弹出最后一个，quit退出程序  **总体思路为：**   1. **利用if判断输入的字符串是否为quit，是则直接return** 2. **如果不是quit，则按照要求操作字符串** 3. **用循环结构，所以函数不需要有返回值，直接打印结果即可**   def checklist(givenlist,oristr):  if oristr.strip()=='':  tmp=[]  tmp.extend(givenlist)  givenlist.pop()  print('{} pop from list'.format(tmp[-1]))  return 0  if oristr=='quit':  print('Goodbye!')  return 0  if oristr in givenlist:  givenlist.remove(oristr)  print('1 instance of {} removed from list'.format(oristr))  return 0  if oristr not in givenlist:  givenlist.append(oristr)  print('1 instance of {} appended to list'.format(oristr))  return 0    return 0  givenlist=['cat', 'goat', 'cat']  while(1):  print('look at all the animals',givenlist)  oristr=input('enter the name of an animal:')  if oristr=='quit':  print('Goodbye!')  break  checklist(givenlist,oristr)  print('\n') 单词检查 Part1 第一部分要求我们完成第一个拼写检查函数，输入两个字符串（不区分大小写）  如果完全相同返回0，长度相同但有一个字母不同返回1，其他返回2  **总体思路为：**   1. **先从简单的判断，都转换为小写字母，如果完全相同，返回0** 2. **对于1的情况，比较长度后遍历对比每一位字母，记录不同字母的数量，如果数量为1则返回1** 3. **其他情况返回2**   def find\_mismatch(str1,str2):  str1=str1.lower()  str2=str2.lower()  if str1==str2:  return 0  if len(str1)==len(str2):  count=0  for i in range(len(str1)):  if str1[i]!=str2[i]:  count=count+1  i=i+1  if count==1:  return 1  else:  return 2 单词拼写Part2 和上一部分类似，只是返回1的判断条件不同，变为通过删减或者增加一个字母可以和另一个单词相同  **总体思路为：**   1. **先从简单的判断，都转换为小写字母，如果完全相同，返回0** 2. **对于1的情况，我们先找出较长的单词（较短的单词可能需要遍历字母表），然后逐个删去里面的一个字母，和另一个单词对比，相同则返回1** 3. **其他情况返回2**   # Type your function here  def single\_insert\_or\_delete(str1,str2):  str1=str1.lower()  str2=str2.lower()  if str1==str2:  return 0  flag=0  if len(str1)<len(str2):  str1,str2=str2,str1  i=0  for char in str1:  resstr=str1[:i]+str1[i+1:]  i=i+1  if resstr==str2:  return 1    else:  return 2 单词拼写Part3   第三部分结合上面两个函数（纠正函数），当返回为1时，利用正确的单词列表替换字符串，达到纠正的效果  **总体思路为：**   1. **首先一定要保证前两个纠正函数没写错** 2. **将字符串处理成列表** 3. **利用if or，如果有任意一个纠正函数为1，则用正确单词替换列表中的单词** 4. **将列表转换回字符串**   # Type your function here  def spelling\_corrector(oristr,correct\_spell):  strlist=list(oristr.lower().split(' '))  res=''  count=0  for word in strlist:  for cword in correct\_spell:  word=word.lower()  cword=cword.lower()  if single\_insert\_or\_delete(word,cword)==1 or find\_mismatch(word,cword)==1:  strlist[count]=cword  break  count=count+1  return ' '.join(strlist)   1. 实验结果分析及回答问题（或测试环境及测试结果）  BMI计算输出字符串  字符串操作拼写检查Part1   这里注意要逐个对比，不能用not in 否则eesy 和 easy，由于有2个e无法正确比较 单词检查Part2  单词检查part3   经过测试，实验结果均和样例相同，说明代码功能正确 |