

《Python 程序设计语言》实验指导书

实验报告说明：

- (1) 实验报告只须提供实验内容和实验总结的答案即可，不用抄题。
- (2) 每次实验后都需提交一份纸质实验报告，并写上姓名和学号。
- (3) 实验报告可打印，也可手写。
- (4) 实验报告可以等到 8 次实验做完后一起提交。
- (5) 实验指导老师：陈峦，13880209111，chluan@uestc.edu.cn，研究院大楼 316#。
- (6) 实验报告请提交到研究院大楼 316#陈峦老师的办公桌上（实验室一直有研究生）。

注：研究院大楼就在西二门旁边，316#实验室在 3 楼上。

实验 3 字符串与正则表达式

一、实验目的

1. 了解字符串编码规则。
2. 掌握字符串的索引与分片。
3. 掌握字符串的操作方法。
4. 掌握正则表达式的构成方法及在字符串处理中的应用。

二、实验内容

请将每个小题的答案写在实验报告中提交。（不用抄题，只提交答案即可）

1. 下面程序的功能是从列表中筛选出字符串元素，并将它们连接为一个字符串。程序运行后的输出情况如下图所示。请在横线处填写适当的语句或表达式将程序补充完整。

AAAUestcBBB

源程序：

```
s=""
```

```
t=[123,'AAA',3.14,'uestc',True,'BBB',456]
```

```
for i in t:
```

```
    if _____:
        s+=i
```

```
else:
```

```
    print(s)
```

2. 下面程序的功能是从键盘读入一个字符串，在文本块中查找该字符串，并输出它在文本块中的位置（用索引下标表示）。程序运行后的输入输出情况如下图所示。请在横线处填写适当的语句或表达式将程序补充完整。

x=bcd
[4, 11]

源程序：

```
x=input('x=')
```

```
s='xyzabcd'*2
```

```
t=[]
```

```

for i in range(len(s)-len(x)+1):
    if x==_____:
        t+= [i]
else:
    print(t)

```

3. 下面程序的功能是将字符串中的子串“bcd”全部替换为字符串“AA”。程序运行后的输入输出情况如下图所示。请在横线处填写适当的语句或表达式将程序补充完整。

```

xyzaAAxyzaAA

```

源程序：

```

x='bcd'
t='AA'
s='xyzabcd'*2
for i in range(len(s)-len(x)+1):
    if x==s[i:i+len(x)]:
        s=s[:i]+t+_____:]
else:
    print(s)

```

4. 下面程序的功能是从键盘读入一个字符串，判断其是否是回文。程序运行后的输入输出情况如下图所示。请在横线处填写适当的语句或表达式将程序补充完整。注：回文是指一个字符串从左向右读与从右向左读相同。例如：noon，level，rotor，boob。

```

s=xyzzzyx
Yes

```

源程序：

```

s=input('s=')
b=True
for i in range(len(s)//2):
    if s[i]!=_____:
        b=False
        break
if b:
    print('Yes')
else:
    print('No')

```

5. 下面程序的功能是从键盘读入一个由阿拉伯数字组成的字符串（中间可以包含一个小数点），如果它是正整数，就删去其左端的所有 0 字符，如果它是正小数，就删去其两端的所有 0 字符，最后再将它转换为数值型数据（int 或 float）。程序运行后的输入输出情况如下图所示。请在横线处填写适当的语句或表达式将程序补充完整。

```

s=00012.340000
12.34

```

源程序：

```

s=input('s=')

```

```

if '.' in s:
    x=float(_____)
else:
    x=int(s.lstrip('0'))
print(x)

```

6. 下面程序的功能是从键盘读入一个英文句子，将其转换为标题格式，即每个单词的首字母大写。程序运行后的输入输出情况如下图所示。请在横线处填写适当的语句或表达式将程序补充完整。

```

s=This is a book
This Is A Book

```

源程序：

```

s=input('s=')
s=s.split()
t=""
for i in s:
    if not i.istitle():
        i=chr(ord(i[0])-32)+_____
    t+=i+' '
else:
    t=t.strip()
    print(t)

```

24. 编写一段程序实现下面的功能：从键盘读入密钥数字和明文，使用凯撒密码算法，将明文加密为密文。

提示：恺撒密码是一种替换加密的技术，明文中的所有字母都在英文字母表上向后（或向前）按照一个固定数目（密钥）进行偏移后被替换成密文。例如，当偏移量是 3（即 `key=3`）的时候，所有的字母 A 将被替换成 D，B 变成 E，以此类推，W 变成 Z，X 变成 A，Y 变成 B，Z 变成 C。

明文	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
密文	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

例如：明文=hello，则对应的密文=khoor。

三、实验总结

完成下面的总结作业题，请将答案写在实验报告上提交。（不用抄题，只提交答案即可）

说明：为便于阅卷，请在实验报告中按下表格式给出单选题的答案。

题号	1	2	3	4	5	6	7	8	9	10
答案										
题号	11	12	13	14	15	16	17	18	19	20
答案										
题号	21	22	23	24	25	26	27	28	29	30
答案										
题号	31	32	33	34	35	36	37	38	39	40
答案										

题号	41	42	43	44	45	46	47	48	49	50
答案										
题号	51	52	53	54	55	56	57	58	59	60
答案										
题号	61	62	63	64	65	66	67	68	69	70
答案										
题号	71	72	73	74	75	76	77	78	79	80
答案										
题号	81	82	83	84	85	86	87	88	89	90
答案										
题号	91	92	93	94	95	96	97	98	99	100
答案										

单项选择题：

1. 运行下面程序段后的输出结果是（ ）

```
s=[]
t=['ABC',{1,2,3},[4,5],['EFG','XYZ']]
for i in t:
    s+=i
else:
    print(len(s))
```

A. 4 B. 5 C. 10 D. 14

2. 运行下面程序段后的输出结果是（ ）

```
s=[]
t=['ABC',{1,2,3},[4,5]]
for i in t:
    s+=i
else:
    print(len(s))
```

A. 3 B. 5 C. 6 D. 8

3. 运行下面程序段后的输出结果是（ ）

```
print(len("\\t\\"101x30u0041\b\n"))
```

A. 8 B. 9 C. 15 D. 24

4. 运行下面程序段后的输出结果是（ ）

```
uestc='成电';成电='电子科技大学';print(eval(uestc))
```

A. uestc B. 成电 C. 电子科技大学 D. '电子科技大学'

5. 运行下面程序段后的输出结果是（ ）

```
s='good good study';s=s.title();print(s)
```

A. Good good study B. 'Good good study'
C. Good Good Study D. 'Good Good Study'

6. 若已知：s='uestc'; 则下列选项中与其余三项的输出结果不同的是（ ）

A. print(s[::-1]) B. print(s[-1::-1]) C. print(s[-1:0:-1]+s[0]) D. print(s[-1:-1:-1])

7. 若已知：s='uestc'; 则下列选项中与其余三项的输出结果不同的是（ ）

A. for i in s[::-1]:print(i,end=") B. for i in s:print(s[len(s)-s.find(i)-1],end=")

- C. `print(s[len(s):-1])` D. `print(s[-1:-len(s):-1])`
8. 若已知: `s='uestc'`; 则下列选项中与其余三项的运算结果不同的是 ()
- A. `s.find('e')` B. `s.rfind('e')` C. `s.index('e')` D. `s.lindex('e')`
9. 下列选项中与其余三项的运算结果不同的是 ()
- A. `'aB'.swapcase()` B. `'aB'.capitalize()` C. `'aB'.title()` D. `'aB'.upper()`
10. 在字符串 `s` 中, 若 $0 \leq i < \text{len}(s)$, 则下列各项中与 `s[i]` 字符相同的是 ()
- A. `s[len(s)-i]` B. `s[i-len(s)-1]` C. `s[i-len(s)]` D. `s[i-len(s)+1]`
11. 在字符串 `s` 中, 若 $0 \leq i < \text{len}(s)$, 则下列各项中与 `s[-i]` 字符相同的是 ()
- A. `s[len(s)-i]` B. `s[len(s)-i-1]` C. `s[i-len(s)]` D. `s[len(s)-i+1]`
12. 表达式: `'23'*1+2*'45'+*67'*0` 的值是 ()
- A. `'234545'` B. `'232345454567'` C. `234545` D. `232345454567`
13. 下列选项中与其余三项的运算结果不同的是 ()
- A. `'abc'.ljust(5, '*')` B. `'**'.join(['abc', ''])` C. `'abc'.join('**')` D. `'abc'.join('(', '**')`
14. 下列选项中与其余三项的运算结果不同的是 ()
- A. `'abc'.join('**')` B. `'abc'.center(5, '*')`
- C. `'**'.join('abc')` D. `('abc'.zfill(4)+'0').replace('0', '*')`
15. 若已知: `t='0120'`; 则表达式: `t.lstrip('0') if t.startswith('0') else t` 的值是 ()
- A. `'0120'` B. `'120'` C. `'012'` D. `'12'`
16. 若已知: `s='012.30'`; 则表达式: `s.rstrip('0') if s.endswith('0') else s` 的值是 ()
- A. `'012.30'` B. `'12.30'` C. `'012.3'` D. `'12.3'`
17. 下列表达式中与其余三项的值不同的是 ()
- A. `'et' in 'uestc'` B. `'ab' > 'ab'` C. `len('abc')==len('""abc""')` D. `str(3.14)==repr(3.14)`
18. 若已知: `s='10203040'`; 则表达式: `s.replace('0', '', s.count('0')-2)` 的值是 ()
- A. `'1234'` B. `'1203040'` C. `'123040'` D. `'12340'`
19. 若已知: `s='10203040'`; 则表达式: `s.split(sep='0', maxsplit=2)` 的值是 ()
- A. `['1', '2', '3', '4', '']` B. `['1', '2', '3040']` C. `['10203', '4', '']` D. `['1020', '3', '4']`
20. 若已知: `s='10203040'`; 则表达式: `s.partition('0')` 的值是 ()
- A. `('1', '0', '203040')` B. `('1', '2', '3040')`
- C. `('1', '0', '2', '0', '3', '0', '4', '0')` D. `('1', '2', '3', '4')`
21. 已知: `s='10203040'`; 若想得到: `'1234'`; 则不正确的表达式是 ()
- A. `"".join([i for i in s if i!='0'])` B. `s.strip('0')`
- C. `"".join(s.split('0'))` D. `s.replace('0', '')`
22. 若已知: `s='uestc'`; 则下列表达式中与其余三项的值不同的是 ()
- A. `s.isalnum()` B. `s.isalpha()` C. `s.islower()` D. `s.isdigit()`
23. 若已知: `s='2A3B'`; 则下列选项中与其余三项的运算结果不同的是 ()
- A. `s.isalnum()` B. `s.isupper()` C. `s.istitle()` D. `s.isalpha()`
24. 下列选项中, 属于 `bytes` 类型的是 ()
- A. `r'\101'` B. `'\101'` C. `0b101` D. `b'101'`
25. 若已知: `s=b'abcd'`; 则下列表达式中与其余三项的值不同的是 ()
- A. `[i for i in s]` B. `list(s)`
- C. `[ord(i) for i in str(s)]` D. `[ord(i) for i in s.decode()]`
26. 下列选项中, 可以用作字典关键字的数据类型是 ()
- A. `list` B. `set` C. `bytearray` D. `bytes`
27. 若已知: `s='uestc'`; 则下列表达式中与其余三项的值不同的是 ()

- A. [(i,s[i]) for i in range(5)] B. [(eval(i[0]),i[1]) for i in zip('01234',s)]
 C. list(enumerate(s)) D. [(i,j) for i,j in zip(b'01234',s)]
28. 若已知: s='uEstC'; 则与下面表达式功能相同的是 ()
 ".join([chr(ord(i)+32) if i.isupper() else chr(ord(i)-32) for i in s])
 A. s.swapcase() B. s.title() C. s.capitalize() D. s.upper()
29. 若已知: s='1020304'; 则与下面表达式功能相同的是 ()
 ".join([i if i!='0' else '-' for i in s])
 A. s.replace('-', '0') B. s.replace('0', '-') C. s.swap('-', '0') D. s.swap('0', '-')
30. 运行下面程序段后的输出结果是 ()

```
x='uestc'
y='UESTC'
t="".maketrans(x,y)
s='uEsTC'
z=s.translate(t)
print(z)
```

 A. UESTC B. uestc C. Uestc D. UeStc
31. 若已知: s='1020304'; 则与下面表达式功能相同的是 ()
 s.translate("".maketrans('0','-'))
 A. s.replace('-', '0') B. s.replace('0', '-') C. s.swap('-', '0') D. s.swap('0', '-')
32. 若已知: s='abcdaxyz'; 则表达式: s.translate("".maketrans('abcd','1234'))的值是 ()
 A. '1234axyz' B. '12341xyz' C. 1234axyz D. 12341xyz
33. 在标准库 string 中, 由 26 个小写英文字母组成的字符串常量是 ()
 A. string.ascii_lowercase B. string.ascii_uppercase
 C. string.ascii_letters D. string.punctuation