

1. 将一个大写字母字符串中的每个字母按 ASCII 码顺序前移 key 个位置（循环排列），然后逆序输出得到密文。其中 key 为随机产生的 4 个[1,9]之间的整数，key 小于字符串长度时可以循环使用。编写 Python 程序代码及实现效果如下：

```
import random
s=input('请输入大写字母字符串：')
k=key=miw=""
for j in range(4):
    k=k+str(_____)
print('随机生成的密钥：'+k)
for i in range(len(s)):
    zm=s[i]
    key=int(k[i%len(k)])
    t=_____
    miw=_____
print("加密后的密文："+miw)
```

请输入大写字母字符串：ABCDE  
随机生成的密钥：9113  
加密后的密文：VABAR

2. 有千位和百位未知的 5 位数正整数“9\*\*65”，能被 37 或 67 整除。下列 Python 程序的功能是：在满足条件的所有数中，将最大一个 5 位数的千位和百位依次显示出来。

```
i,flag=100,False
while i>0 and not flag:
    _____
print(i//10,i%10)
```

方框中的代码由以下三部分组成：

- ①j=90065+i\*100      ②i-=1      ③if (j%37)\*(j%67)==0:flag=True

下列选项中，代码顺序正确的是（ ）

- A. ①②③      B. ①③②      C. ②①③      D. ②③①

3. 有如下 Python 程序段：

```
s="CixiStudent"
f=[0]*26
t=s[4:]
i=0
while i<len(t):
    zm=ord(t[i])-ord('a')
    if "A"<=t[i]<="Z":
        i+=1
        continue
    elif "a"<=t[i]<="z" and f[zm]==0:
        f[zm]+=1
        i+=1
for i in range(26):
    if f[i]==1:
        print(chr(i+ord('a')),end="")
```

执行该程序段后，输出的内容是（ ）

- A. dentu      B. Student      C. tuden      D.deintux

4. 小王最近迷上了踢毽子，他的计划是第一天踢 1 个，第二天和第三天连续 2 天每天踢 2 个，第四天到第六天连续 3 天每天踢 3 个，以此类推。现在小王每天都严格按照进阶计划训练，编写 Python 程序统计截止到第 n 天训练结束时他总共踢了几个毽子，程序运行界面如图所示。请回答下列问题：

(1) 按照小王的训练计划, 第 8 天他必须踢\_\_\_\_\_个键子。

(2) 请在划线处填入合适的代码。

```
n=int(input("请输入训练天数: "))
sumx=0 #当前训练的踢键子总数
t=0    #当前训练的总天数
x=0
while t<=n:
    x+=1
    _____
    sumx+=x**2
sumx=_____
print('您在',n,'天中共踢了',sumx,'个键子')
```

请输入训练天数: 30  
您在 30 天中共踢了 156 个键子

5. 小明为选出班级平均分最高的参赛作品编写了一个 Python 程序, 设计如下:

(1) 5 位评委对 3 个作品的评分数据存储在列表 data 中 (数据由 3 位十进制数组成, 第 1 位对应作品编号, 后 2 位对应作品得分, 如 “275” 表示 2 号作品得分 75)。

(2) 统计输出 3 个作品的平均分, 选择平均分最高的作品作为参赛作品, 若最高平均分存在并列, 则输出全部并列的最高平均分作品。

程序运行界面如图所示, 请在划线处填入合适的代码。

```
data=[180,283,385,170,276,384,180,285,380,190,295,390,170,272,372]
```

```
s=[0,0,0] #存储 3 个作品的得分
```

```
ans=[] #存储并列最高平均分的作品号
```

```
maxb=0
```

```
for i in range(len(data)):
```

```
    zp=_____ #分离出作品编号
    fs=data[i]%100 #分离出作品得分
    _____ #累加当前作品的得分
```

```
for j in range(3):
```

```
    print('作品',j+1,'平均分为: ',s[j])
```

```
    if s[j]>maxb:
```

```
        maxb=s[j]
```

```
for z in range(3): #查找并列最高平均分
```

```
    if _____:
```

```
        ans.append(z+1) #将数据添加到列表 ans 尾部
```

```
print('平均分最高的作品号是: ',ans)
```

作品 1 平均分为: 78.0  
作品 2 平均分为: 82.2  
作品 3 平均分为: 82.2  
平均分最高的作品号是: [2, 3]

6. 某 Python 程序实现功能为: 在第一行输入一个仅包含字符 “0” 和 “1” 的字符串 (要求字符串中没有连续的 “1”), 将其中任意位置的 “0” 翻转为 “1”, 但要确保翻转后的字符串中仍然没有连续的 “1”, 求最多可翻转的 “0” 的个数并输出。程序实现效果和代码如下, 请在划线处填入合适的代码。

```
s='0'+input()+'0'
```

```
c=0;i=2
```

```
n=len(s)-1
```

```
while i<n:
```

```
    if _____:
```

```
        i=i+2
```

```
    elif s[i-1]=='1':
```

```
        i=i+1
```

100101000100  
最多可翻转 2 个

```
elif _____:
```

```
    i=i+3
```

```
else:
```

```
    _____  
    c=c+1
```

```
print('最多可翻转',str(c),'个')
```

7. 小江编写 Python 程序完成不同进制的加法运算。输入一个由二进制、十进制或十六进制组成的加法式子，求出这个加法式子的和（用十进制表示）。加法式子形式如“A2H+1101B=”，程序运行界面如图所示。

实现上述功能的程序如下，请回答下列问题：

```
def xtod(s,m):    #将 m 进制数 s 转换为十进制数
```

```
    n=len(s);y=0
```

```
    for k in range(0,n):
```

```
        ch3=s[k]
```

```
        if "F">=ch3>="A":
```

```
            x=ord(ch3)-55
```

```
        else:
```

```
            x=int(ch3)
```

```
        y=_____
```

```
    return y
```

```
s1=input("请输入式子:")
```

```
i=0;ans=0
```

```
dic={"B":2,"D":10,"H":16}
```

```
for j in range(0,len(s1)):
```

```
    ch1=s1[j]
```

```
    ch2=s1[j-1]
```

```
    if ch1=="+" or ch1=="=":
```

```
        s2=s1[i:j-1]
```

```
        ans=_____
```

```
print("运算结果为:"+str(ans)+"D")
```

(1) 若输入的加法式子为“1BH+10B+65D”，则它们的和（十进制数）为\_\_\_\_\_。

(2) 在程序划线处填入合适的代码。

8. 计算圆周率  $\pi$  的方法有很多，其中“蒙特卡罗算法”又称作“随机数法”是其中之一。如图 1，在边长为 1 的正方形 ABCD 中有一扇形 BAD，在正方形 ABCD 内随机生成  $n$  个点（大小忽略不计），我们只需统计出在扇形 BAD 内的点数  $ns$  和在正方形 ABCD 内的点数  $n$  就可以计算出圆周率的近似值，公式如图 2

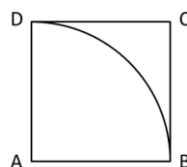


图 1

$$\frac{ns}{n} \approx \frac{S_{\text{扇}}}{S_{\text{正}}} = \frac{\pi/4}{1} = \frac{\pi}{4}$$

图 2

我们需要生成在正方形中的点  $(x,y)$  来进行模拟（其中  $0 \leq x, y \leq 1$ ），用  $x^2 + y^2 < 1$  来表示落在扇形内部的点。实现上述功能的 Python 代码如下，请在划线处填入合适的代码：

```
import random
```

```
n=100000 # 产生 n 个点
```

```
ns=0
```

```
def judge(x,y):
```

```
    flag = False
```

```
    if _____ < 1:
```

```

        flag = True
    return flag
i=1
while _____:
    x=random.uniform(0, 1) #产生 [0,1] 范围内的实数
    y=random.uniform(0, 1)
    if _____:
        _____
    i+=1
print("圆周率的近似值 :",4*ns/n)

```

9. 查找与替换。从键盘上分别输入要查找和替换的字符串，对文本文件进行查找与替换，替换后保存到新的文本文件中。完成查找与替换功能的思路是：首先可从待检索文本文件“in.txt”逐行读取文本内容到列表 text，然后从键盘上输入查找的字符串 key 和替换的字符串 new，对列表 text 中的元素逐个进行查找并替换，结果保存到列表 result，最后将 result 写入文件“out.txt”。

```

text = readfile("in.txt") #读入文件
key = input("请输入要查找的字符串：")
new = input("请输入要替换的字符串：")
result = []
for line in text:
    newline = replace(key, new, line) #替换
    result.append(newline) #添加到列表
writefile("out.txt",result) #写入文件

```

```

def readfile(filename):
    #逐行读取文本文件数据存入 text 列表并返回，代码略
def writefile(filename,text):
    #将 text 写入 filename 文件，代码略

```

(1) 查找字符串，如下的 findstr 函数，在字符串 line 中从 begin 位置开始查找 key 在字符串 line 中的位置，请在划线处填入合适的代码。

```

def findstr(key,line,begin):
    for i in range(begin, len(line) - len(key) + 1):
        if _____:
            return i
    return -1

```

(2) 替换字符串。如下的 replace 函数，在字符串 line 中检索所有的字符串 key 并替换为 new，请在划线处填入合适的代码。

```

def replace(key,new,line):
    begin = 0
    while begin < len(line) - len(key) + 1:
        pos = findstr(key,line,begin)
        if pos == -1:
            _____
        else:
            line = line[0:pos] + new + line[pos + len(key) :len(line)]
            begin = pos + len(key)
    return line

```