例 4.1 Python 程序代码

```
import itertools
from itertools import permutations, combinations_with_replacement
import fractions
import sys

# 4.2 (1)
a = (1, 2, 3)  # 三个球
b = ('A', 'B', 'C', 'D')  # 四个杯子
# 样本空间基本事件总数
num = 3 ** 4
# 从四个球中选择两个球投入第一个杯子里
ite = itertools.combinations(b, 2)
lite = list(ite)
# 再将剩下的两个球投入另一个杯子
b1 = ('A', 'B') or ('A', 'C') or ('A', 'D') or ('B', 'C') or ('B', 'D') or ('C', 'D')
it_e = itertools.combinations(b1, 2)
lit_e = list(it_e)
print(lit_e)
p = fractions.Fraction(len(lite)*len(lit_e), num)
print(p)
sys.exit()

# 4.1 (1)
res = []
for i in range(1, 5):
    for j in range(1, 5):
        for k in range(1, 5):
            res.append(i*100+j*10+k)
print(res)  # 样本空间中所有的事件数

result=[]
for i in range(1,5):
    for j in range(1,5):
        for k in range(1,5):
            if (i+j+k)%3 == 0:
                result.append(i*100+j*10+k)
print(result)  # 样本空间中能被 3 整除的事件数
P = len(result)/len(res)
print(f'{len(result)}/{len(res)}')  # 输出概率结果

# 4.1(2)
bag1 = [1, 2, 3]
c = list(itertools.permutations(bag1, 2))
```

```
d = list(itertools.combinations(bag1, 2))
print(d)
print(len(c))
p = fractions.Fraction(len(d), len(c))
print(p)
sys.exit()

# 4.1 (3)
bag_2 = [1, 1, 2, 2, 2]
ite = list(itertools.permutations(bag_2, 3))
print(len(ite))
print(ite)
count = 0
for item in ite:
    if item == (1, 2, 2):
        count += 1
print(count)
p = fractions.Fraction(count, len(ite))
print(p)
sys.exit()

# 4.1 (4)
bag5 = [1, 1, 2, 2, 3, 3, 4, 4, 5, 5]
ite2 = list(itertools.combinations(bag5, 4))
print(ite2)
print(len(ite2))
for i in bag5:
    if i == 1:
        num1 = len(list(itertools.combinations([2, 2, 3, 3, 4, 4, 5, 5], 1)))
num2 = len(list(itertools.combinations([3, 3, 4, 4, 5, 5], 1)))
num = len(list(itertools.combinations(range(1, 6), 1)))
s = (num * num1 *num2)
print(s)
p = fractions.Fraction(s, len(ite2))
print(p)
```