The following 13 hours long YouTube program: "**Python for Everybody - Full University Python Course"** is worthwhile in watching again and again. I learned quite a lot from it myself.

$ Supplement Note 2

Problem 1.4 *How many ways of distributing 4 identical black balls and 4 identical white balls into 6 distinct boxes?*

Solution: The distribution of black balls and white balls are independent events. The total number of ways of distributing 4 identical black balls into 6 distinct boxes *is equivalent* to that of arranging 4 identical black balls and 5 identical "bars" (*to separate 6 boxes!*), which is given by $\frac{(4+5)!}{4!5!} = \binom{9}{4}$. Hence the answer is given by $\binom{9}{4}\binom{9}{4} = \left(\frac{9\cdot 8\cdot 7\cdot 6}{4!}\right)^2$.

Factorials and Stirling's Formula

The *factorial function* $n! \equiv n\cdot(n-1)\ldots 2\cdot 1$ is a rapidly increasing function of n, as demonstrated in the python code *t2.py*. In this respect, a A very powerful approximation for the factorial function is provided by **Stirling's formula**, whose derivation is given in section **8.6.3**, p. 8-25:

$$n! \approx \sqrt{2\pi n}\,\frac{n^n}{e^n} \tag{8.45}$$

Even for $n = 1, 2, 3$, the results from Eq.(8.45), $0.92214, 1.9190, 5.8362$, are already good approximations for the exact values $1, 2, 6$. The values of $n!$ for $n = 1, \ldots, 20$ are calculated and plotted in the python code *t2.1.py* and are exhibited in Table 1 at the end of the note. The following supplement problem demonstrates the power of Stirling's formula:

Supplement Problem 1 *Calculate the probability that no two students in a class of 50 students have the same birthday.*

Solution: The probability is clearly given by $\binom{365}{50}\frac{1}{365^{50}}$, which may be simplified by using Stirling's formula

$$\binom{365}{50}\frac{1}{365^{50}} = \frac{365!}{315!\,50!}\frac{1}{365^{50}} \approx \frac{\sqrt{2\pi\cdot 365}\,\left(\frac{365}{e}\right)^{365}}{\sqrt{2\pi\cdot 315}\,\left(\frac{315}{e}\right)^{315}\,\sqrt{2\pi\cdot 50}\,\left(\frac{50}{e}\right)^{50}}\frac{1}{365^{50}}$$
$$= \frac{\sqrt{365}\,\left(365\right)^{315}}{\sqrt{2\pi\cdot 315\cdot 50}\,\left(315\right)^{315}\left(50\right)^{50}}$$

This result is calculated in python code *t2.1.py* and is equal to $9.758\ 10^{-67}$. Due to the rapid increase of the magnitudes of the factorial function, a direct calculation can easily blow up if the order of product is not carefully arranged!

The list of t2.py and t2.1.py is given here.

```
# t2.py: factorial
import numpy as np
import matplotlib.pyplot as plt
factorial = 1
nn = np.arange(1,11)
fac = []
with open('factorial10.txt','w') as fout:
    fout.write('n, n!=\n')
    for n in range(1, 11):
        factorial *= n
        fac.append(factorial)
        print('n, n!=', n, factorial)
        fout.write(str(n) + ', ' + str(factorial)+ '\n')
plt.plot(nn,fac,label='factorial')
plt.xlabel('n')
plt.legend()
plt.show()
## t2.1.py: factorial
import numpy as np
import matplotlib.pyplot as plt
Pi2  = 2 *np.pi  # 2 pi
```

```
s = np.sqrt(365/( Pi2*315*50))* (365/315)**315/50**50
st = 'The probability that there are no identical birthdays among 50 students'
st += " is\n {0:7e} \n".format(s)
print(st)
fac = 1   # factorial
x, y1, y2, yerr = [], [], [], []
with open('factorial.txt','w') as fo2:
    fo2.write("n, n!, n! from Stirling's formula, error =\n")
    for n in range(1, 21):
        x.append(n)
        fac *= n
        y1.append(fac)
        c = np.sqrt(Pi2*n) * (n/np.e)**n     # Stirling's formula
        y2.append(c)
        efac = c/fac-1   #error  or Stirling's formula
        yerr.append(efac)
        s = str(n) + ', ' + "{:.4e}".format(fac)+', '
        s += "{:.4e}".format(c)+ ', '+ str(efac) + '\n'
        ss = "{0:4d},  {1:.4e}, {2:.4e} , {3: 5.2e}\n".format(n,fac,c,efac)
        fo2.write(ss)
plt.plot(x,y1,linestyle = 'dashed', label='factorial')
plt.plot(x,y2,linestyle = 'dotted', label="factorial from Stirling's formula")
plt.plot(x,yerr,label='error')
#plt.title('Factorial')
plt.xlabel('n')
plt.legend()
plt.show()
```

n	n!	Stirling's formula	error
1	1.0000e+00	9.2214e-01	-7.79%
2	2.0000e+00	1.9190e+00	-4.05%
3	6.0000e+00	5.8362e+00	-2.73%
4	2.4000e+01	2.3506e+01	-2.06%
5	1.2000e+02	1.1802e+02	-1.65%
6	7.2000e+02	7.1008e+02	-1.38%
7	5.0400e+03	4.9804e+03	-1.18%
8	4.0320e+04	3.9902e+04	-1.04%
9	3.6288e+05	3.5954e+05	-0.92%
10	3.6288e+06	3.5987e+06	-0.83%
11	3.9917e+07	3.9616e+07	-0.75%
12	4.7900e+08	4.7569e+08	-0.69%
13	6.2270e+09	6.1872e+09	-0.63%
14	8.7178e+10	8.6661e+10	-0.59%
15	1.3077e+12	1.3004e+12	-0.55%
16	2.0923e+13	2.0814e+13	-0.51%
17	3.5569e+14	3.5395e+14	-0.48%
18	6.4024e+15	6.3728e+15	-0.46%
19	1.2165e+17	1.2111e+17	-0.43%
20	2.4329e+18	2.4228e+18	-0.41%

Table 1: factorial from 1 to 20