南京大学《智能程序设计（Python 语言）》

2025 秋

SymPy&NumPy 练习参考答案

编程题

1.

```
>>> from sympy import *
>>> x = symbols('x')
>>> expr = x*ln(sqrt(2)+1/x)
>>> limit(expr, x, +oo)
 Oo
```

2.

```
x = np.random.normal(loc=5, scale=5**0.5, size=1000)
print("max: %.3f" % x.max())
print("min: %.3f" % x.min())
print("avg: %.3f" % x.mean())
print("std: %.3f" % x.std())
print("q1: %.3f" % np.quantile(x, 0.25))
print("q2: %.3f" % np.quantile(x, 0.5))
print("q3: %.3f" % np.quantile(x, 0.75))
print(x.dtype, x.itemsize*x.shape[0])
x = x.astype('int32')
print(x.dtype, x.itemsize*x.shape[0])
```

3.

```
A = np.random.uniform(-10, 10, 5*4).reshape(5, 4)
# A = np.random.uniform(-10, 10, (5, 4))
print(A)
flat_A = A.reshape(-1)
ind = flat_A.argsort()
loc = np.append(ind[:2], ind[-2:])
flat_A[loc] = 0
A = flat_A.reshape(A.shape)
print(A)
print("(4,6) number", np.sum((A>4) & (A<6)))
```

4.

```
def func(x):
    return sum(map(int, str(x)))

sum_array = np.frompyfunc(func, 1, 1)
arr = np.arange(1, 1001)
print("vectorized sum: ",  sum_array(arr), sep = '\n')
```