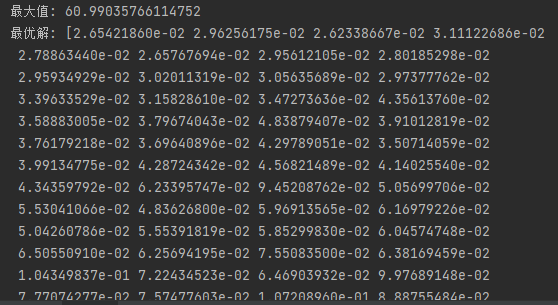
**数模第三次上机**

# 第一题

答案：  


代码：

import numpy as np  
from scipy.optimize import minimize  
  
# 定义目标函数  
def objective(x):  
 return -np.sum(np.sqrt(x))  
  
# 定义约束条件  
def constraint1(x):  
 return 10 - x[0]  
  
def constraint2(x):  
 return 20-(x[0] + 2\*x[1])  
  
def constraint3(x):  
 return 30-(x[0] + 2\*x[1] + 3\*x[2])  
  
def constraint4(x):  
 return 40 - (x[0] + 2\*x[1] + 3\*x[2] + 4\*x[3])  
  
def constraint5(x):  
 return 1000 - np.sum([(101-i)\*x[i-1] for i in range(1 ,101)])  
  
def constraint6(x):  
 return [x[i] for i in range(100)]  
  
# 定义约束条件类型  
constraint\_type = [{'type': 'ineq', 'fun': constraint1},  
 {'type': 'ineq', 'fun': constraint2},  
 {'type': 'ineq', 'fun': constraint3},  
 {'type': 'ineq', 'fun': constraint4},  
 {'type': 'ineq', 'fun': constraint5},  
 {'type': 'ineq', 'fun': constraint6}]  
  
# 定义初始猜测值  
x0 = np.zeros(100)  
  
# 求解最大化目标函数的问题  
solution = minimize(objective, x0, method='SLSQP', constraints=constraint\_type)  
  
# 打印结果  
print("最大值:", -solution.fun)  
print("最优解:", solution.x)

# 第二题

答案：



代码：

import numpy as np  
from scipy.optimize import minimize  
  
def objective(x):  
 return -(2\*x[0] + 3\*x[0]\*\*2 + 3\*x[1] + x[1]\*\*2 + x[2])  
  
def constraint1(x):  
 return 10-(x[0] + 2\*x[0]\*\*2 + x[1] + 2\*x[1]\*\*2 + x[2])  
  
def constraint2(x):  
 return 50- (x[0] + x[0]\*\*2 + x[1] + x[1]\*\*2 - x[2])  
  
def constraint3(x):  
 return 40 - (2\*x[0] + x[0]\*\*2 + 2\*x[1] + x[2])  
  
def constraint4(x):  
 return x[0]\*\*2 + x[2] - 2  
  
def constraint5(x):  
 return (x[0] + 2\*x[1]) - 1  
  
# 定义初始点  
x0 = np.array([0,0,0])  
  
# 定义约束  
cons = [  
 {'type': 'ineq', 'fun': constraint1},  
 {'type': 'ineq', 'fun': constraint2},  
 {'type': 'ineq', 'fun': constraint3},  
 {'type': 'eq', 'fun': constraint4},  
 {'type': 'ineq', 'fun': constraint5}  
]  
  
bounds = ((0,None) , (None,None), (None,None))  
  
# 最小化求解  
result = minimize(objective, x0, constraints=cons , method='SLSQP')  
print("最大值" , -result.fun)  
print("最优解" , result.x)