Plat使用手册

1. 问题类

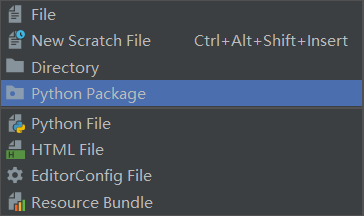
定义问题时需要继承PROBLEM类，并且将问题放在Problems文件夹下。

每一个问题都有以下属性以及函数组成，如下所示：

|  |  |
| --- | --- |
| 属性 | |
| N | 种群规模 |
| D | 决策变量维度 |
| M | 目标个数 |
| encoding | 编码方式：’real‘，’permutation‘ |
| parameter | 问题相关参数 |
| FE | 当前评估次数 |
| maxFEs | 最大评估次数 |
| 函数 | |
| Initialization | 初始化初代种群 |
| CalObj | 计算适应度值 |

具体操作如下：

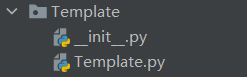
首先，新建Python Package



并将Package的名字修改成问题的名字，此时，Package中将自动生成一个\_\_init\_\_.py。



接着，在已生成的Package中新建python file，文件名为问题名。



每个问题需要继承PROBLEM类，并重新定义 CalDec，例如SOP\_F1.py：

1. from Problems.PROBLEM import PROBLEM
2. import numpy as np
3. class SOP\_F1(PROBLEM):
4. # Initialization of the problem
5. def \_\_init\_\_(self):
6. super().\_\_init\_\_()
7. # Calculate the objective of a solution
8. def CalDec(self, PopDec):
9. PopObj = np.sum(PopDec \*\* 2, 1)
10. return PopObj

各行代码功能如下：

第一行：表示继承PROBLEM类

第二行：导入numpy类

第四行：定义问题SOP\_F1并继承PROBLEM

第六行：类的构造函数

第七行：继承父类\_\_init\_\_()函数

第九行：定义CalObj，用于计算适应度值

第十行：计算适应度值

第十一行：返回适应度值

2. 算法类

2. 个体类

SOLUTION对象用于表示一个个体，一组SOLUTION对象可以表示整个种群。个体类的属性以及函数如下所示：

|  |  |
| --- | --- |
| 属性 | |
| Problem | 解所求解的问题 |
| PopDec | 解的决策变量 |
| PopObj | 解的适应度值 |

该类不需要用户定义，可以按照如下方式进行使用：

1. SOLUTION(np.ramdom.random((100,5)))

以上代码表示生成一个种群规模为100\*5的种群。