以下是对这些代码文件及其主要功能的简要介绍：

### **1. cropgro\_strawberry\_implementation.py**

此文件包含了 Cropgro 草莓生长模拟模型的核心实现，主要功能如下：

* simulate\_day 函数：模拟草莓一天的生长过程，根据天气数据更新物候期、计算光合作用、蒸腾作用、水分胁迫等，并更新植物的生物量和状态，最后将当天的模拟结果存储在 results 列表中。
* run\_example\_simulation 函数：运行一个示例模拟，生成模拟所需的土壤属性、品种参数和天气数据，初始化模型并运行模拟，最后绘制模拟结果。

### **2. cropgro-strawberry-test1.py**

该文件是 Cropgro 草莓模型的测试文件，包含多个测试用例：

* simulate\_day 函数：与 cropgro\_strawberry\_implementation.py 中的 simulate\_day 功能相同，用于模拟一天的草莓生长。
* simulate\_growth 函数：按顺序对天气数据集中的每一天运行模拟，并将结果存储在 results\_df 中。
* update\_fruits 函数：在开花和坐果阶段添加果实到植株上。
* 多个测试用例：如 test\_single\_day\_simulation 测试单天模拟，test\_full\_simulation 测试完整生长模拟，test\_plotting 测试绘图功能，test\_model\_inputs\_outputs 测试模型的输入输出等。
* run\_test\_suite 函数：运行测试套件并打印测试结果。

### **3. show\_dataframe\_details.py**

此文件主要用于读取天气数据、运行 Python 模型、读取 DSSAT 输出并进行比较：

* parse\_dssat\_date 函数：将 DSSAT 的 YYDDD 日期代码转换为 YYYY - MM - DD 字符串。
* read\_wth\_file 函数：解析 DSSAT 的.WTH 文件并将其转换为 DataFrame。
* read\_fortran\_output 函数：读取 DSSAT 生成的 PlantGro.OUT 文件。
* run\_python\_model 函数：使用 Python 模型进行生长模拟。
* main 函数：读取天气数据，运行 Python 模型，读取 DSSAT 输出（如果存在），并比较两者的结果，最后将模拟结果保存为 CSV 文件。

### **4. run\_all\_comparisons.py**

该文件用于对所有草莓实验进行验证：

* run\_validation 函数：对单个实验文件进行验证，调用 validate\_models.py 脚本并保存验证报告。
* main 函数：查找所有草莓实验文件，对每个文件运行验证，并输出验证总结。

### **5. explain\_row\_differences.py**

此文件解释了 Python 模型和 DSSAT Fortran 模型输出行数不同的原因：

* main 函数：详细阐述了两种模型在模拟方法、输出频率、目标用户、计算效率和数据使用等方面的差异，并说明如何进行模型验证。