频繁项集 Apriori 算法实验要求和内容

一、 实验目的

1. 熟悉、理解购物篮模型，理解并掌握 Apriori 算法过程
2. 能够编程实现 Apriori 算法，并能计算一些简单实例

二、 实验环境和工具

最好使用 Python 3 版本，能够用其他程序语言实现也可以

三、 算法过程：

1) 输入：购物篮数据集，支持度阈值
2) 初始化 C1，得到所有的单项集
3) 扫描一遍购物篮，结合支持度阈值，得到 Ck 过滤后的 Lk
4) 构造通过 Lk 构造下一步所需的 C(k+1)
5) 重复上面 3,4 两步，直到没有频繁项出现
6) 输出所有频繁项

四、 实验过程

参照课本第 156 到 157 页内容，算法核心见图 6-4

1. 定义构造 C1 的函数，完成初始化。储存方式可以设定为购物篮以二维数组储存数据，且数据都是数字，其代码对应为：

```
def create_c1(date_set):
    c_1 = set([])
    for item in date_set:
        c_1 = c_1.union(set(item))
    return [frozenset([i]) for i in c_1]  # 因为后面要用它作为字典的元素，所以需要用 frozen
形式
```

2. 定义构造 Lk 的函数，即完成算法中的过滤过程，对应的代码为：

```
def get_lk(date_set, c_k, support_threshold):  # 建立 Lk 的函数
    l_k = {}
    for item in date_set:  # 将整个字典扫描一遍
        for c_i in c_k:
            if c_i.issubset(item):
                if c_i not in l_k:
                    l_k[c_i] = 1
                else:
                    l_k[c_i] += 1
    l_k_return = []  # 用于返回的集合 l_k
```

```
    for l_i in l_k:
        if l_k[l_i] >= support_threshold:
            l_k_return.append(l_i)
    return l_k_return
```

3. 建立构造 C(k+1)的函数：该步骤由同学们自行完成。如果完成程度不是很好的话，我会在之后给出代码供同学们参考。
4. 以上函数建立好之后，直接在主函数中调用函数，并通过循环完成频繁项集的计算工作，最后输出所有的频繁项集
5. 输入所用到的实例如下：

```
[[1, 3, 4], [2, 3, 5], [1, 2, 3, 5], [2, 5]]
```

支持度阈值设为 2 即可。