1.任务介绍：给哈佛峰会的选课分课结果，写一个“排课表”的程序脚本，希望能用于今年以及以后每一年重复使用，希望可以用最简单的语言（例如excel VBA）写，并能让以后不懂程序的人也能掌握如何跑这个脚本，达到工作效率的最大化。

一、排课表算法任务描述：

问题描述：共有650-700左右学生。约一半在北京上课，一半在上海。北京共有25-30门课，上海也是。上海和北京上课是完全分开的，学生也完全分开，互相不干扰。夏令营中，每门课重复开班4次（时段1 - 时段4），迎来4批完全不同的学生。每一个时段中，全部的课程都是同时上课的。每一个时段里，每一个学生都在上他的4门课里的其中一门。在夏令营前，每个学生在看完所在会场所有课程的描述后，根据兴趣填写8个志愿，8个志愿全部来自他所在的那一个会场所开设的课程里，不会选另外一个会场的课程。志愿1是他最想上的，以此类推，一直到志愿8。现在，通过一个已有的脚本，每个学生已经被分配了4门互相不重复的课(course 1- course 4，这仅代表根据每门课的最大可容纳人数限制而分配的课程结果，而不代表每个学生在4个时段上课的先后顺序)。

需要做的这个新算法的任务，是需要给每个学生定出他4门课上课的顺序（即每个人的课程表，每个学生的course 1 - course 4上课的先后顺序，使得一门课的4个时段里人数相对平均）。要求是：每个时段的每门课人数控制在12个人左右（原则上不能比10个少或者比14个多，除非总共填报这门课志愿的人就过于少）。希望每一门课的每一个时段里，人数尽量平均，尽量不出现一门课在时段1里人非常多，但是时段2、3、4人很少这样的情况。

二、如何理解原始数据：

附件里是往届的志愿和选课数据，作为例子，用来试写算法。

1. 选课记录：

附件的“hsylc\_regist\_record”表里，一行代表一个学生的选课志愿。rgr\_userID是学生的唯一ID，rgr\_ID是用户选课填志愿记录的ID，也就是说可能有多于一条的rgr\_ID的rgr\_userID可能是重复的，需要在处理时去掉重复的。假设我们只考虑每一个学生重复选课的最后一次选课的志愿，也就是只考虑rgr\_ID最大的那一次记录。

2. 志愿：

每一个选课记录里，rgr\_regist\_course1到rgr\_regist\_course8代表着一个学生的8个志愿，1是最喜欢的，以此类推。志愿是课程ID，课程ID的全部列表可以看“stats”表里面的B列。C列代表着课程所在的会场，10代表北京会场，21代表上海会场。

3. 分课结果：

每一个选课记录里，rgr\_final\_course1到rgr\_final\_course4代表着一个学生的4个分课结果，4个分课结果一定都是来自8个志愿里的，除了极个别情况，因为一门课已经爆满（上限是一个时段14个学生，即总共56个学生，如果排到一个学生时，他的全部8个志愿的课程都已经超过了这个上限，那么有可能给他force一门不是他志愿的课，也就是P列里面的forceCourse记录的被强制分配的课程。

三、需要算法达到的效果：

1. 产出：在“hsylc\_regist\_record”表里，对于每一条选课记录，根据算法得出新的4个column的数据，里面的数字代表学生在时段1-时段4中分别应该上的课程ID。也就是最终的“课程表”。我们叫做“课程A-课程D”

2. 准确性：课程A-课程D应该互相不重复。课程A-课程D应该和rgr\_final\_course1-rgr\_final\_course4完全相同，只是顺序不同。

3. 平均性：对于任何一个课程ID来说，所有在课程A时段上这门课的学生总数，和B，C，D时段上课的学生总数，应该相对平均，人数差希望不超过4。

4. 合理性：对于任何一个课程ID来说，所有在课程A时段上这门课的学生总数，和B，C，D时段上课的学生总数，应该都处于合理的课堂人数范围，最高最好不超过15，最小最好不小于7

5. 可便捷重复：希望最终的脚本，是一个可以简单用excel重现、每年重复使用的脚本。不知道这次会用什么语言来写，希望不要用太复杂的语言，不需要每次运行都在需要特殊配置的计算机环境。最理想，是用excel VBA来写，这样以后每一年只需要重新运行一次VBA就可以直接在excel里得到结果了。

6. 可直观交接给别人：因为每年这个脚本应该需要设置几个参数，比如“最大人数差”、“每堂课最小人数”、“每堂课最大人数”等等，希望能简单说明以后如何由别人来设置这些参数。最好能让以后没有编程知识的人，也能轻易地掌握如何设置参数、如何跑脚本，而不需要每次都由程序员做。

2．Demi,

收到，经过检查，完全符合要求，而且结果很好，干得漂亮！

等你考完AP，能否接着看一下我以下这个邮件里提到的“算法A”的任务，也就是也用VBA重新写一个给学生分配四门课的小程序，代码结构上最好能和算法B分开来。写完之后，我们可以对比一下你的算法A和原来很多年前写好的算法A的结果哪个更优化。目标：（1）尽可能多的给学生分配top choice，越多越好。（2）尽可能少地给学生强制分配不属于志愿里的课程，最好是0。出现强制分配的原因，一般是因为有些课过于火爆，如果一个学生他的8个志愿全都是非常火爆的，那么很有可能等脚本运行到他的时候，发现不管给他分配什么课程，都会导致那门课程超出最大上限，于是只能强制分课。我想象你可以通过提前计算每门课的填志愿火爆程度来避免这种情况的发生。

北京会场一门课的最大人数，理论上等于北京会场总人数除以北京会场总课程数。上海会场同理。但是对于非常火爆的课程，可能会需要超过最大上限。理论上，一门课人数在12左右比较健康。在极端情况下，最多到15人，还是可以接受的。每门课程重复上4次，所以对于算法A，人数上限都要乘以4。

目前这个表里没有直接标出每个学生是哪个会场，你可以通过match一下这个学生的课程志愿号是属于北京课程还是上海课程。建议能在第一张表的rgr\_userId右侧再加一列，用10和21表示学生的会场，这样以后我们可以把这张表的结构作为标准的运行数据的表格结构，以后的工作人员只需要把需要的行列的数据填写好，然后运行你的脚本，就可以直接得出结果，不需要更多手工了。

最后就是，请帮助看一下，这个excel表格的结构可以怎样优化，目标是以后每年只需要把rgr\_id, rgr\_userId，8个志愿，还有北京和上海的课程ID列表数据放入这个excel模板，就可以自动运行程序了。因为每年人数和课程数目都不一样，希望最后整理好的模板是可以不受到具体人数和课程数量的影响的（例如，把tab 2里面的北京和上海课程列表拆分成两张表，可能是一个更干净的做法，给你参考）

谢谢！