**Deadline：2017-09-30（周日）晚上11:00**

## 1题目：实现一个自动生成小学四则运算题目的命令行程序。

## 2说明：

自然数：0, 1, 2, …。

* 真分数：1/2, 1/3, 2/3, 1/4, 1’1/2, …。
* 运算符：+, −, ×, ÷。
* 括号：(, )。
* 等号：=。
* 分隔符：空格（用于四则运算符和等号前后）。
* 算术表达式：

e = n | e1 + e2 | e1 − e2 | e1 × e2 | e1 ÷ e2 | (e),

其中e, e1和e2为表达式，n为自然数或真分数。

* 四则运算题目：e = ，其中e为算术表达式。

## 3需求：

1. 使用 -n 参数控制生成题目的个数，例如

Myapp.exe -n 10

将生成10个题目。

2. 使用 -r 参数控制题目中数值（自然数、真分数和真分数分母）的范围，例如

Myapp.exe -r 10

将生成10以内（不包括10）的四则运算题目。该参数可以设置为1或其他自然数。该参数必须给定，否则程序报错并给出帮助信息。

3. 生成的题目中计算过程不能产生负数，也就是说算术表达式中如果存在形如e1 − e2的子表达式，那么e1 ≥ e2。

4. 生成的题目中如果存在形如e1 ÷ e2的子表达式，那么**其结果应是真分数**。

**5. 每道题目中出现的运算符个数不超过3个。**

6. 程序一次运行生成的题目不能重复，**即任何两道题目不能通过有限次交换+和×左右的算术表达式变换为同一道题目**。例如，23 + 45 = 和45 + 23 = 是重复的题目，6 × 8 = 和8 × 6 = 也是重复的题目。**3+(2+1)和1+2+3这两个题目是重复的，由于+是左结合的，1+2+3等价于(1+2)+3，也就是3+(1+2)，也就是3+(2+1)。但是1+2+3和3+2+1是不重复的两道题，因为1+2+3等价于(1+2)+3，而3+2+1等价于(3+2)+1，它们之间不能通过有限次交换变成同一个题目。**

生成的题目存入执行程序的当前目录下的Exercises.txt文件，格式如下：

1. 四则运算题目1

2. 四则运算题目2

……

其中真分数在输入输出时采用如下格式，真分数五分之三表示为3/5，真分数二又八分之三表示为2’3/8。

7. 在生成题目的同时，计算出所有题目的答案，并存入执行程序的当前目录下的Answers.txt文件，格式如下：

1. 答案1

2. 答案2

特别的，真分数的运算如下例所示：1/6 + 1/8 = 7/24。

8. 程序应能支持一万道题目的生成。

9. 程序支持对给定的题目文件和答案文件，判定答案中的对错并进行数量统计，输入参数如下：

Myapp.exe -e <exercisefile>.txt -a <answerfile>.txt

统计结果输出到文件Grade.txt，格式如下：

Correct: 5 (1, 3, 5, 7, 9)

Wrong: 5 (2, 4, 6, 8, 10)

其中“:”后面的数字5表示对/错的题目的数量，括号内的是对/错题目的编号。为简单起见，假设输入的题目都是按照顺序编号的符合规范的题目。

## 4、博文规范

1. 在文章开头给出**Github项目地址**。（1分）
2. 在开始实现程序之前，PSP表格(参加附录1)记录下你估计将在程序的各个模块的开发上**耗费的时间**。（1分）
3. **效能分析，**记录你在改进程序性能上花费了多少时间，描述你改进的思路，并展示一张性能分析的图。如果可能，展示你程序中消耗最大的函数。（3分）
4. **设计实现过程**。设计包括代码如何组织，比如会有几个类，几个函数，他们之间关系如何，关键函数是否需要画出流程图？（5分）
5. **代码说明**。展示出项目关键代码，并解释思路与注释说明。（4分）
6. **测试运行**。共享你对程序进行测试的至少10个测试用例，以及说明为什么你能确定你的程序是正确的。（3分）
7. 在你实现完程序之后，在PSP表格记录下你在程序的各个模块上**实际花费的时间**。（1分）
8. **项目小结**。结对项目总结成败得失，分享经验，总结教训。两个人共同撰写一个博客，包含上述内容的描述，同时包含**结对感受**，以及两个人对彼此结对中的**闪光点或建议的分享**。（2分)

## 5、评分规则

本次个人项目分数由两部分组成，分别是

1. 博客 — 20分，分数组成在博文规范中。
2. 程序 — 40分（附加分10分）
   * 5分为源代码管理评分，该评分主要通过源代码管理中的commit注释信息，增量修改的内容，是否有运行说明等给分。
   * 5分为代码具有规范性。
   * 10分为正确性评分，按要求输入参数后，程序能够正确运行。
   * 20分为功能评分，完全满足需求1-8。
   * 附加分10分，实现需求9得10分
3. **注意事项：**

* 按时间完成并提交——正常评分
* 晚交一周以内——0分
* 晚交一周以上或不交——倒扣本次作业分数
* **抄袭——倒扣2倍本次作业分数【严禁代码与博客等一切形式的抄袭！博客园支持了对博客的查重功能，我们也有专用的代码查重系统进行代码查重。请各位同学千万不要触碰底线，勿谓言之不预也！】**

## 附录

### 1. PSP2.1表格

| **PSP2.1** | **Personal Software Process Stages** | **预估耗时（分钟）** | **实际耗时（分钟）** |
| --- | --- | --- | --- |
| Planning | 计划 |  |  |
| · Estimate | · 估计这个任务需要多少时间 |  |  |
| Development | 开发 |  |  |
| · Analysis | · 需求分析 (包括学习新技术) |  |  |
| · Design Spec | · 生成设计文档 |  |  |
| · Design Review | · 设计复审 (和同事审核设计文档) |  |  |
| · Coding Standard | · 代码规范 (为目前的开发制定合适的规范) |  |  |
| · Design | · 具体设计 |  |  |
| · Coding | · 具体编码 |  |  |
| · Code Review | · 代码复审 |  |  |
| · Test | · 测试（自我测试，修改代码，提交修改） |  |  |
| Reporting | 报告 |  |  |
| · Test Report | · 测试报告 |  |  |
| · Size Measurement | · 计算工作量 |  |  |
| · Postmortem & Process Improvement Plan | · 事后总结, 并提出过程改进计划 |  |  |
| 合计 |  |  |  |