

**信息与软件工程学院**

**综合设计II中期报告**

课程名称： 综合设计Ⅱ

课题名称： 银行信用卡业务后台子系统的设计与实现

指导教师： 许毅

所在系别： 软件工程（大型主机方向）

执行学期： 大三

学生信息：

|  |  |  |
| --- | --- | --- |
| 序号 | 学号 | 姓名 |
| 1（组长） | 2014220402030 | 杨林彬 |
| 2 | 2014220402033 | 罗阳星 |
| 3 | 2014220401008 | 郑伟 |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |

目 录

第一章 综合设计的进展情况 1

1.1 针对工程问题的方案设计 1

1.2 针对工程问题的推理分析 2

1.3 针对工程问题的具体实现 4

1.4 知识技能学习情况 5

第二章 存在问题与解决方案 6

2.1 存在的主要问题 6

2.2 解决方案 6

第三章 前期任务完成度与后续实施计划 10

参考文献 14

# 第一章 综合设计的进展情况

针对工程问题的方案设计

我们选择的课题名称为银行信用卡业务后台子系统的设计与实现。经过仔细的调研，我们对于银行信用卡的基本业务流程有了一定的了解。



图-1 日常生活中的信用卡

[1]银行信用卡（Credit Card）是银行向个人和单位发行的，可以凭此向特约单位购物、消费和向银行存取现金，其形式是一张正面印有发卡银行名称、有效期、号码、持卡人姓名等内容，由芯片、磁条、签名条组成的磁卡。持卡人持信用卡消费时无需支付现金，待账单日时再还款。

信用卡业务中的关键词：

1. 账单日：银行每月会固定一天对持卡人的信用卡账户中当期发生的各项交易、费用等进行汇总结算，并结记利息，计算持卡人当期总欠款金额和最小还款额，打印出对账单邮寄给客户以便持卡人进行核对，并提示持卡人还款的最后日期和还款金额。
2. 到期还款日：发卡行要求持卡人归还当期信用卡应付款项的最后日期，一般为对账单日起的第20天。
3. 全额还款：持卡人再信用卡到期还款日前偿还对账单上所列示的全部应付款项。采取全额还款的持卡人其当期对账单所列示的信用卡消费可以享受免息还款的待遇。
4. 最低还款：持卡人在到期还款日前偿还当期对账单上的全部应付款项有困难时，可以按发卡行规定的最低还款额进行还款，如持卡人选择最低还款，就不能享受免息还款的待遇。
5. 免息还款期：持卡人利用信用卡进行交易消费，从银行记账日起至到期还款日的期间，在此期间，持卡人只要全额还清当期对账单上所列示的全部应付款项，便不用向银行支付消费交易所产生的贷款利息。

结合信用卡业务和实际开发情况，我们对于本次项目在功能上提出了以下的目标：

1. 系统能够模拟用户交易的数据
2. 数据文件将会囊括消费、取现、还款三大客户行为
3. 程序处理数据文件，生成账单，即统计文件。

我们的计划是使用本学期新学习的一门脚本语言REXX，来模拟用户的交易数据，把模拟出来的数据通过FTP上传到主机上，然后用COBOL语言实现业务的后台逻辑。

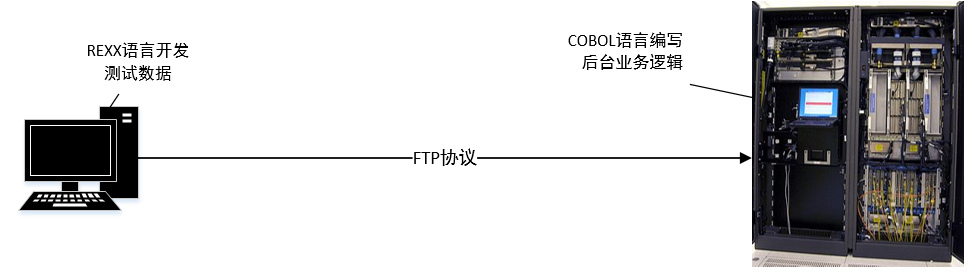


图-2 初步方案

针对工程问题的推理分析

这个项目从表面上来看是不复杂的，主要就是使用COBOL语言实现信用卡的后台业务逻辑。但是软件作为一种工具，它存在的价值不是在于其本身开发的复杂程度，而是要看该软件是否很好地对现实业务进行了模拟。所以前期我们地主要精力没有放在实际开发上面而是着重于银行卡后台业务的调研工作。我们以下面这个例子来解释以下银行卡的具体业务过程：

【例】Alice的账单日为每月18日，到期划款日为每月的7日。4月18日银行为Alice打印的本期账单包括了他从3月19日到4月18日之间的所有交易业务。本账单周期Alice仅有一笔消费：在4月15日消费人名币1000元。在4月18日的账单上显示Alice要还金额为1000元，如果Alice在5月7日前偿还100元，计算5月18日对账日显示的利息？

解：具体的计算如下

这个例子只是针对消费的过程，但是根据我们设计指导书上面的要求是要模拟用户取现的，因此对于取现我们查阅了资料，发现取现的过程不同于消费，从取现日期开始一个周期内，每天都按照0.05%的利率计算利息。所以我们在实际开发的过程中应该要把利息分成两类：

1. 消费利息
2. 取现利息

在预估了系统的功能实现方案后，我们对于系统的结构进行了一个大致的规划，制作了下面的这个系统总用例图：

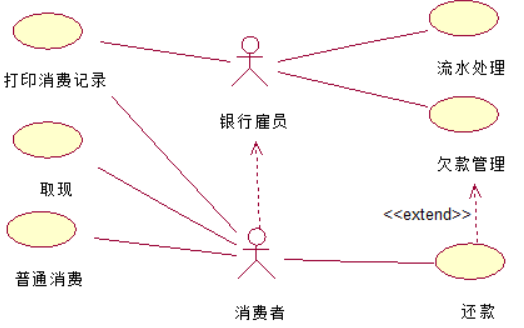


图-3 系统总用例图

我们在前期和老师做了多次交流，我们这个项目主要是使用COBOL编写批处理脚本程序，要把精力集中在对于信用卡业务的研究和逻辑实现上，更加深入的研究与开发会在综合设计Ⅲ中来展开。

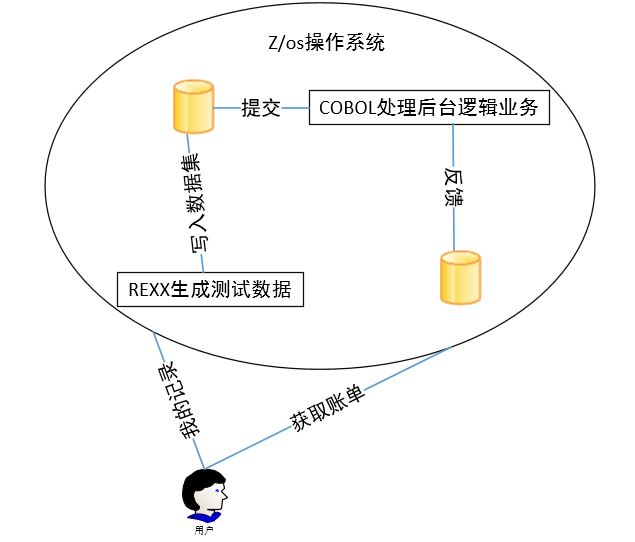


图-4 系统大致框架

针对工程问题的具体实现

正如前面文档中提到的，我们的测试数据是使用REXX开发的。信用卡后台业务逻辑打算使用COBOL语言来开发。我们之所以选择REXX来作为测试数据的开发工具主要有以下几点考虑：

1. REXX是一种解释型语言，风格比较简约，上手容易
2. REXX语言风格特殊，写出来的代码非常易懂，便于大家传阅交流

交易数据设计的格式为：

表格1-1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 交易号 | 持卡人 | 交易数额 | 交易日期 | 交易类型 | 银行卡额度 |
| 日期+持卡人代号+随机数 | 持卡人代号：  XR+序列号 | 为一个正整数 | 格式为：  YYYYMMDD | 1. 消费 2. 取现 3. 还款 | 额度值：5000 |

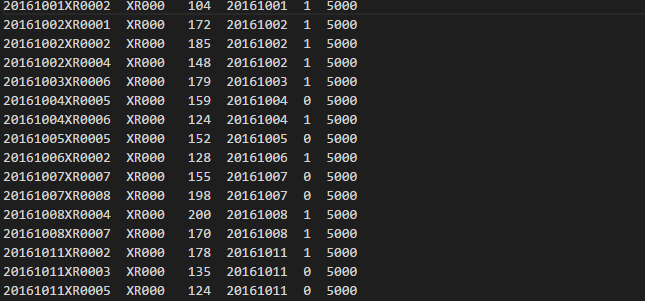
使用REXX模拟的情况：

图-5 REXX模拟用户交易数据

知识技能学习情况

前期还没有涉及到COBOL代码的开发，主要是对于信用卡业务的调研以及REXX语言的学习。

对于REXX，我们主要还是在windows平台上开发。后期随着我们学习的深入，会考虑在主机的Z/OS操作系统上进行开发，这样就省去了FTP传输文件的步骤，我们的开发工作也会高效得多。主机开发和windows平台下的开发还是有很大不同的，举个例子，写入文件（数据集），在windows平台上使用的是call lineout命令而在主机上则是用EXECIO命令。所以我们还需要对REXX深入的学习，目前生成的数据以后更具实际开发的需要可能还会做出调整。

# 第二章 存在问题与解决方案

2.1 存在的主要问题

在开发过程中主要遇到的问题：

一、缺少开发环境：

该银行信用卡后台业务子系统需要在Z/OS系统中开发，由于大型主机机房开发时间有限，所以需要自行搭建并配置Z/OS虚拟机环境以便进行开发。

二、盘卷已满导致无法启动TSO：

在编写程序时由于COBOL程序无限循环导致盘卷已满，此时无法再次登入系统进行操作。

* 1. 解决方案

1. 开发环境搭建[2]：

IBM主机系统可以借助仿真软件Hercules 在PC上面运行。在安装主机系统之前，必须遵循IBM对于主机软件的有关规定。

1. 准备并安装相关软件和文件：

主机系统卷文件，PCOM 3270仿真终端，Hercules，  Hercules CTCI-W32等TCP/IP支持软件。

为了方便管理，约定以下几个路径，存放z/OS模拟环境所需的各项文件和程序。

表2-1 虚拟机文件路径

|  |  |
| --- | --- |
| D:/ADCDV1R6 | 所有与模拟环境相关的东西都放在这个目录下 |
| D:/ADCDV1R6/ZOSV1R6 | 存放卷文件，即CCKD文件 |
| D:/ADCDV1R6/Config | 存放启动配置文件 |
| D:/ADCDV1R6/HercGUI-1.11.1 | 存放Hercules GUI和FishLib |
| D:/ADCDV1R6/hercules-3.07 | 存放Hercules的可执行文件和CTCI-W32 |
| D:/ADCDV1R6/Log | 存放系统运行日志 |

2、配置Hercules启动配置文件zos1.9.cnf:

在输入配置文件时，需要注意以下几个字段，

- LOADPARM是z/OS在IPL时需要用到的参数，参数选择不同，z/OS中启动后，运行的组件也不同。  
 - MAINSIZE表示Hercules占用的物理内存大小，单位是MB，如果内存够大，最好设为1024，在这里暂且设为832，建议不要小于512，否则MIPS会很低，系统运行会非常的慢。   
 - NUMCPU表示虚拟的主机有多少个CPU。   
 - Display Terminals 表示3270终端的数量，演示所用的3270终端号是从0700开始的，以十六进制记录，如0700-0710则表示16个端口，其中0700是控制台专用端口，0701至0710为用户连接端口。   
 - DASD Device，要把所有下载的CCKD文件都列进去，这一段的每一行分为三个部分，第一个是设备编号，如0A80，第二个是设备类型，如3390，第三个是卷文件的路径和文件名，这里可以写绝对路径也可以是相对路径。

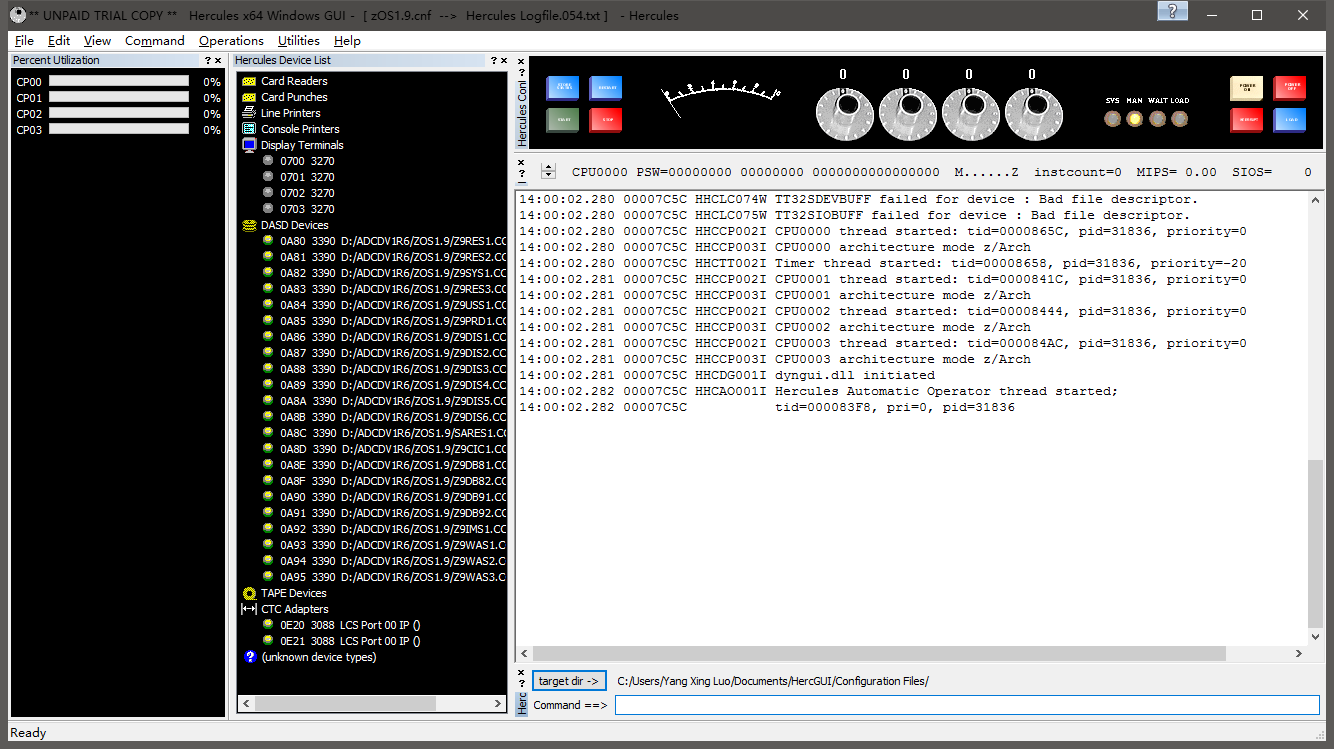


图-66 Hercules仿真软件

3、启动虚拟机系统：

打开Hercules仿真软件，选择配置文件加载盘卷文件，再打开一个PCOM仿真终端窗口，再在Hercules启动虚拟机，此时在第一个PCOM窗口进行初始化操作，当操作完成后就可以打开一个新的PCOM窗口，就可以登录Z/OS系统了。

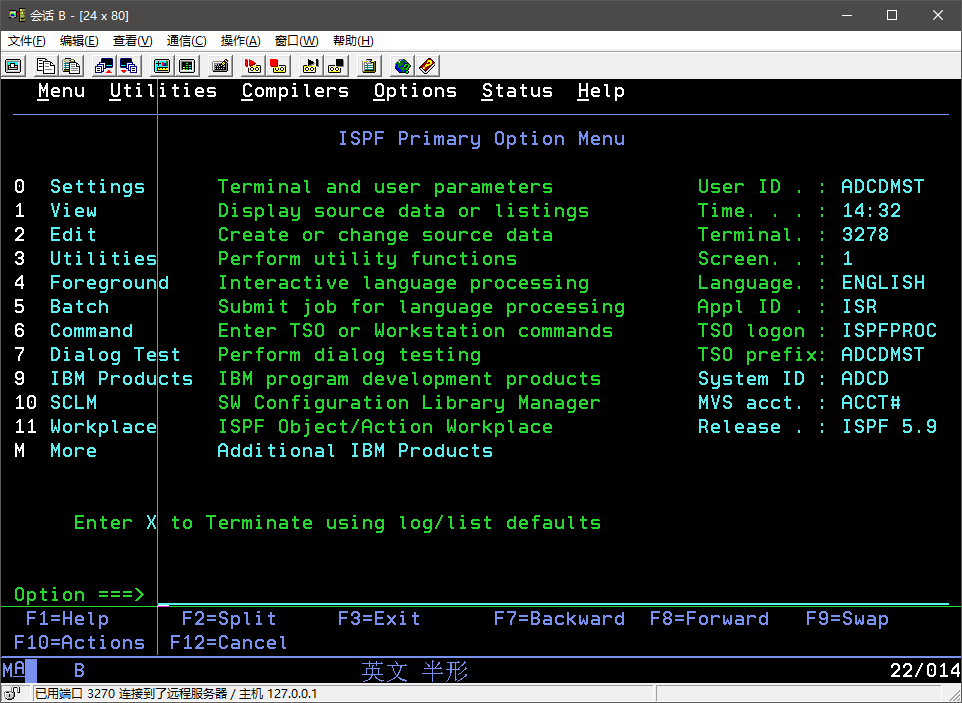


图-7 启动后的虚拟机

4、配置COBOL编译器：

虚拟机启动后，还需要配置COBOL编译器，在编写编译COBOL的JCL时，需要在调用编译器的语句之前指明编译器的位置

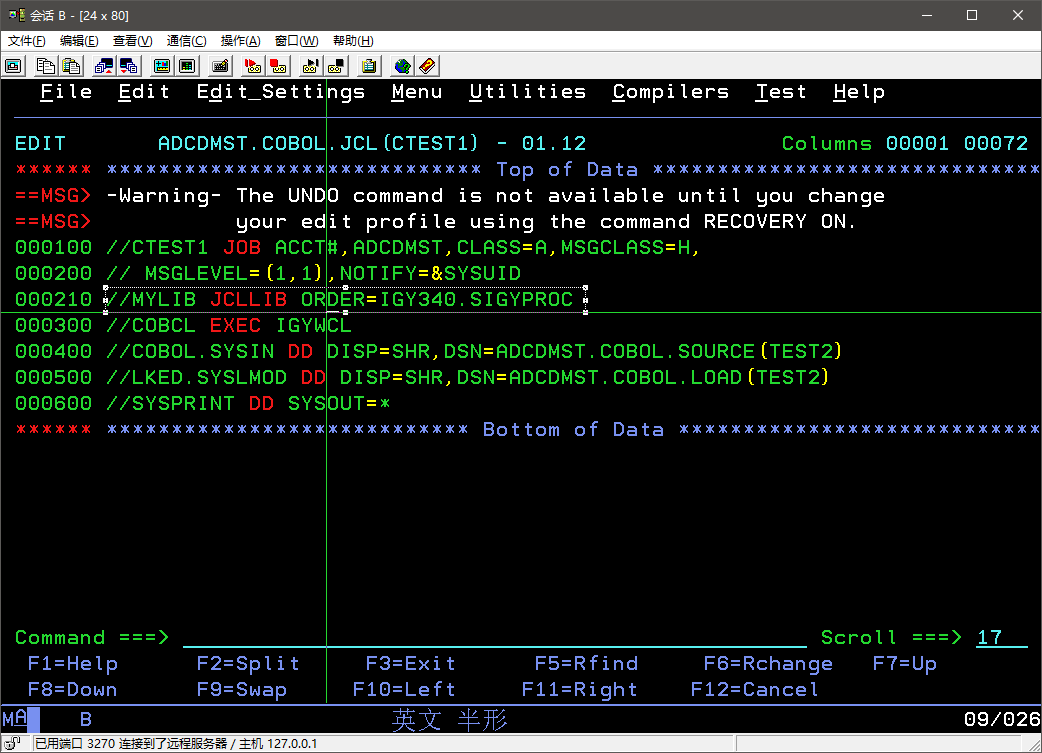


图-8 配置编译器的位置

二、冷启动虚拟机[3]：

Hercules的LOADPARM参数指明了虚拟机的启动方式，其形式为0A82XXM1,其中XX可以有很多形式，32即代表冷启动，重新设置参数后，启动大机，就可以再次登入系统。

冷启动时需要根据提示回复信息：R 0，COUPLE = \*\*。

# 第三章 前期任务完成度与后续实施计划

3.1 前期任务完成度

前期的工作主要是围绕着信用卡的业务调研和开发环境的搭建和配置上面开展的。就像前面的报告提到的那样，本次项目从实际开发来说难度是不大的，可是，一个软件产品的好坏并不是从开发技术复杂度来判断的，而是在于该软件是否能够对现实世界的实体进行很好的建模模拟。我们前期的调研分析实际上就是在为后期COBOL后台业务逻辑编码做准备。

经过前期的分析设计与初步的实现，学习了REXX语言以及COBOL语句在项目关键部分的使用，现在我们已经大致设计出了交易数据的格式，搭建并配置好了Z/OS操作系统的虚拟机环境，后期会持续跟进，尽快写出COBOL的demo出来，因为考虑到最终我们还会根据实际对程序进行不断的调优。

前期团队的具体分工如下表3-1所示

表3-1 前期团队分配工作情况表

|  |  |  |
| --- | --- | --- |
| 团队成员 | 前期主要任务 | 角色及责任 |
| 杨林彬 | 分配工作任务，组织需求与设计的讨论，设计信用卡消费逻辑，使用COBOL进行逻辑的预实现 | 组织、设计、编码 |
| 罗阳星 | 参与信用卡消费逻辑的设计，使用COBOL语句完成数据集的读写操作预实现 | 编码、调试 |
| 郑伟 | 参与信用卡消费逻辑的设计，使用REXX语句完成测试数据的生成与调试 | 编码、调试 |

完成的测试消费记录数据的生成代码（部分）如下图3-1（a）、3-1（b）、3-1（c）所示：

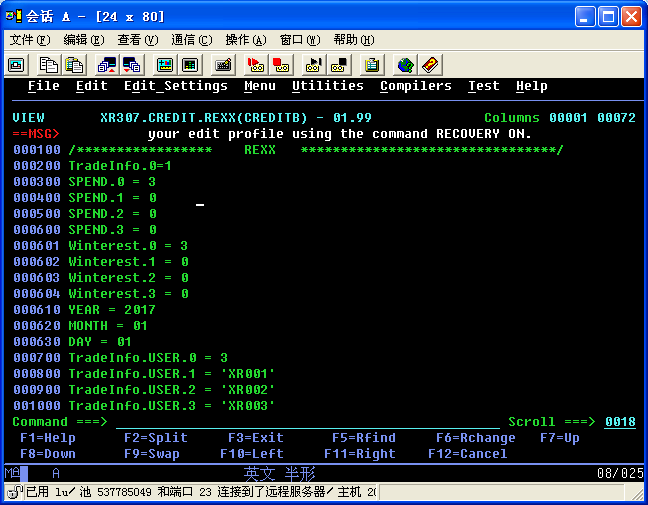
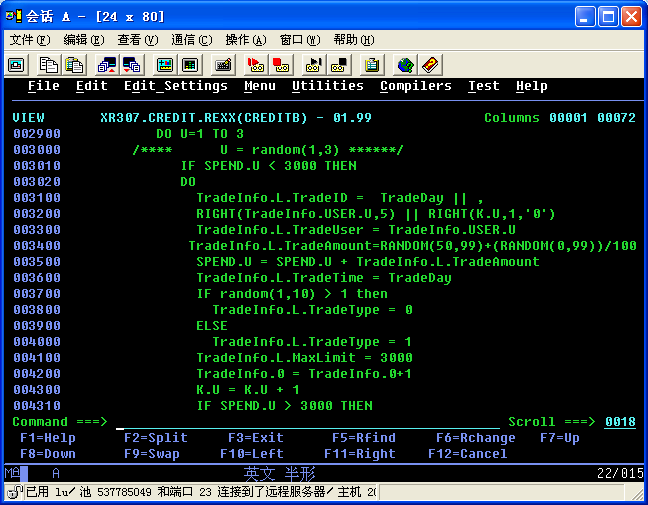


图3-1（a） 生成测试数据代码

图3-1（b） 生成测试数据代码

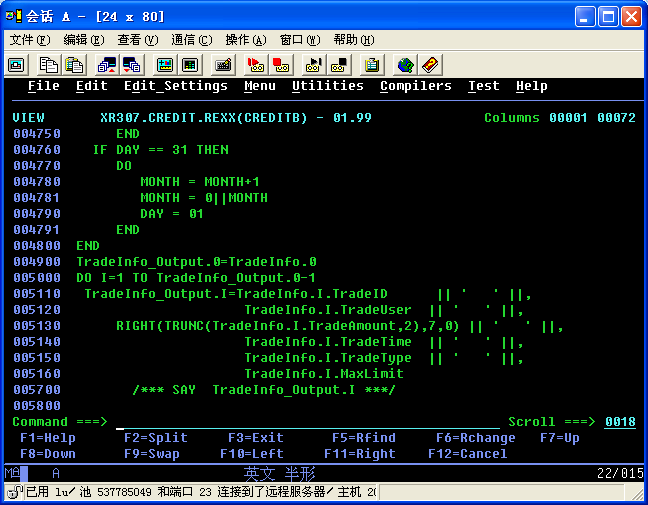


图3-1（c） 生成测试数据代码

生成的数据格式也是符合预期设计的要求，具体如下图3-2所示：

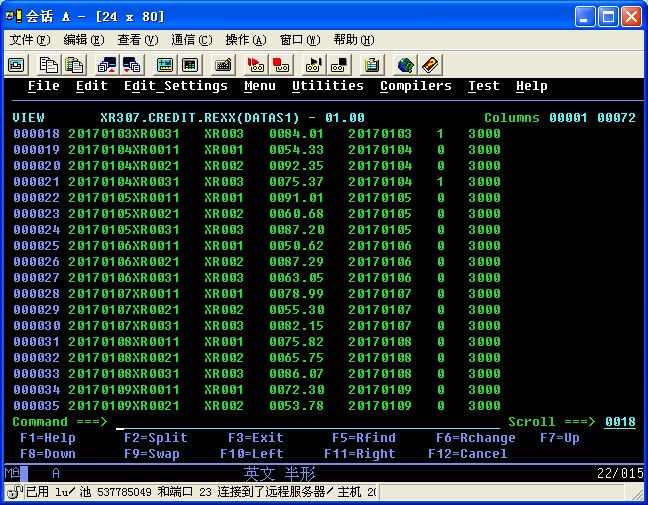


图3-2 生成的初步交易数据格式

3.2 后续实施计划

根据前期任务完成度的情况，项目主要后续工程环节主要包括以下阶段：

项目详细实现阶段：使用COBOL语言对消费记录进行完整的读取，并根据所设计的利息计算逻辑进行各用户的利息计算，最后生成账单。

项目测试阶段：使用多种生成的数据对系统进行测试，观察输出结果，完成测试报告。

项目结题阶段：根据前期完成任务以及后期计划实施的情况完成项目的最终报告。

接下来的时间，我们的具体开发计划如下：

1. 利用空闲时间，小组成员一起研究如何生成更为有效和规则的数据格式，并在大机上编译通过运行。
2. 在12月中旬之前设计出COBOL的demo程序，结合测试数据实现基本的功能需求。
3. 月底到答辩前的工作主要围绕着软件产品性能的调优和结题报告书来展开，会给出系统的详细需求分析书和开发手册。

# 参考文献

[1]信用卡基本常识[[EB](https://www.baidu.com/s?wd=EB&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YLn1nsuARsmHNBnWF9myDd0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWcvP101njcLnjRsP1mLPWDYr0)/[OL](https://www.baidu.com/s?wd=OL&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YLn1nsuARsmHNBnWF9myDd0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWcvP101njcLnjRsP1mLPWDYr0)].(2016-11-25). http://jingyan.baidu.com/article/4ae03de31.html

[2]修正zos安装配置启动错误[[EB](https://www.baidu.com/s?wd=EB&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YLn1nsuARsmHNBnWF9myDd0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWcvP101njcLnjRsP1mLPWDYr0)/[OL](https://www.baidu.com/s?wd=OL&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YLn1nsuARsmHNBnWF9myDd0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWcvP101njcLnjRsP1mLPWDYr0)].(2016-11-25). <https://zhidao.baidu.com/question/118208.html>

[3]大型机在Windows下环境的搭建[[EB](https://www.baidu.com/s?wd=EB&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YLn1nsuARsmHNBnWF9myDd0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWcvP101njcLnjRsP1mLPWDYr0)/[OL](https://www.baidu.com/s?wd=OL&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YLn1nsuARsmHNBnWF9myDd0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWcvP101njcLnjRsP1mLPWDYr0)].(2016-11-25). http://blog.csdn.net/tuliangde/article/details