Infosys Hashers 2012 – Round 1 – Problem Explanation:

Teams will be given two files

- plans.txt (this contains the set of billing plans from mobile service providers)
- usage.txt (this contains the set of monthly usage data for many months for a consumer)

The task is to write a program that, from the two files:

1 – Identifies the cheapest plan for each usage data (UsageSet), and

2 – Identifies the cheapest plan for the entire set of usage data (all UsageSet).

Input file 1: usage.txt – attached

Input file 2: plans.txt – attached

Competition Environment – will be available from 12 noon IST Thursday 25 October 2012 in batches till 12 noon IST Sunday 28 October 2012.

Each activity in the competition environment will be logged. Any evidence of tampering will result in disqualification.

Each participant must work in directory called InfyHash under their home directory.

Each participant must submit a text file along with multiple output files and code.

The text file must be named <College>_<Team_Name>.txt – this file should contain each and every step undertaken to create the environment in which that specific code can run. This is extremely critical, as each and every code will be run in a different instance and the text file must have explicit instructions to recreate the exact environment. This file should also contain the approach, the description of the solution, the choice of technology, and the team's experience in the team's own words.

Input files:

Each program must consume files named plans.txt and usage.txt in the same directory

Output Format:

Each output file must be named <College>_<Team>_YYYYMMDDhhmmss.txt. In case multiple output files are submitted, the one that the team is proud of should be named<College>_<Team>_YYYYMMDDhhmmss_proud.txt

Each team must run the following commands before initiating any session:

```
yum install psacct-6.3.2
chkconfig psacct on
/etc/init.d/psacct start
```

The first executable statement of every program must output this line in the output file

<College> | <Team_Name> | <Team_Member_One_Official_EmailId> | <Team_Member_Two_Official_EmailId> | <System Time>

This <System Time> must be saved as <Start System Time>. Its format should be yyyymmddhhmmss. It is also a part of the output filename

The second executable statement of every program must output this line in the output file

<College> | <Team_Name> | <Start System Time> | StartReadingPlanData | <System Time>

This <System Time> must be in the format yyyymmddhhmmsssss. It should capture seconds and milliseconds

Once the processing of plans.txt has ended, the output should be

<College> | <Team_Name> | <Start System Time> | EndReadingPlanData | <System Time>

The next line should be

<College> | <Team_Name> | <Start System Time> | StartUsageProcessing | <System Time>

This should be followed by the following for every UsageSet

<College> | <Team_Name> | <Start System Time> | <UsageSet> | <CheapestPlan> | <CheapestCost>

Once each UsageSet is over, the output should be

<College> | <Team_Name> | <Start System Time> | EndUsageProcessing | <System Time>

This should be followed by the following for the sum total of all UsageSets

<College> | <Team_Name> | <Start System Time> | StartAllUsageProcessing | <System Time>

<College> | <Team_Name> | <Start System Time> | <AllUsageSets> | <CheapestPlan> | <CheapestCost>

<College> | <Team_Name> | <Start System Time> | EndAllUsageProcessing | <System Time>

E.g. Sample output: Filename – IITK_Anonymous_20121026153437

```
IITK | Anonymous | abc@iitk.ac.in | def@iitk.ac.in | 20121026153437
IITK | Anonymous | 20121026153437 | StartReadingPlanData | 20121026153437123
IITK | Anonymous | 20121026153437 | EndReadingPlanData | 20121026153438456
IITK | Anonymous | 20121026153437 | StartUsageProcessing | 20121026153438457
IITK | Anonymous | 20121026153437 | Anu2008 | RSMin | 1234.56
IITK | Anonymous | 20121026153437 | Anu2009 | CCMin | 987.56
IITK | Anonymous | 20121026153437 | Anu2010 | RSMax | 123.56
IITK | Anonymous | 20121026153437 | Anu2011 | OWMax | 12345.78
IITK | Anonymous | 20121026153437 | EndUsageProcessing | 20121026153439654
IITK | Anonymous | 20121026153437 | StartAllUsageProcessing | 20121026153440123
IITK | Anonymous | 20121026153437 | AllUsageSets | RSMax | 23456.78
IITK | Anonymous | 20121026153437 | EndAllUsageProcessing | 20121026153441123
```

Monthly Charges	CCMax	CCMin	OWMax	OWMin	RSMax	RSMin
	Anu201209	Anu201209	Anu201209	Anu201209	Anu201209	Anu201209
OnNetLocalRsSEC	900	900	900	1499.4	1800	3600
DiscountOnNetLocalSEC	300	375	240		600	
Total OnNetLocalRsSEC	**600**	**525**	**660**	**1499.4**	**1200**	**3600**
OffNetLocalRsSEC	750	1249.5	1000.5	1875	1500	3000
DiscountOffNetLocalSEC			80.04			
Total OffNetLocalRsSEC	**750**	**1249.5**	**920.46**	**1875**	**1500**	**3000**
OnNetSTDRsSEC	1374.45	2750.55	2062.5	4125	1650	3300
DiscountOnNetSTDSEC	499.8				600	
Total OnNetSTDRsSEC	**874.65**	**2750.55**	**2062.5**	**4125**	**1050**	**3300**
OffNetSTDRsSEC	2250.45	2925.45	1687.5	3375	1350	2700
DiscountOffNetSTDSEC						
Total OffNetSTDRsSEC	**2250.45**	**2925.45**	**1687.5**	**3375**	**1350**	**2700**
ISD1RsSEC	**3200.1**	**3200.1**	**3200.1**	**3200.1**	**3200.1**	**3200.1**
ISD2RsSEC	**3063.4**	**3063.4**	**3063.4**	**3063.4**	**3063.4**	**3063.4**
SMSLocalRsSMS	625	2500	750	2500	25	50
DiscountSMSLocalNUM		500	30		5	
Total SMSLocalRsSMS	**625**	**2000**	**720**	**2500**	**20**	**50**
SMSSTDSMS	375	2250	1125	2250	15	30
DiscountSMSSTDNUM			75			
Total SMSSTDRsSMS	**375**	**2250**	**1050**	**2250**	**15**	**30**
DiscountSMSNationaNUM	250					
Total SMSNationaRsSMS	750	4250	1770	4750	35	80
SMSISDRsSMS	**1250**	**1250**	**1250**	**1250**	**2.5**	**5**
SMSRoamingRsSMS	**550**	**550**	**550**	**550**	**5.5**	**11**
RoamingIncomingRsSEC	**749.7**	**1500.3**	**1500.3**	**1500.3**	**900**	**1800**
RoamingOutgoingLocalRsSEC	**1000.2**	**1999.8**	**1999.8**	**1999.8**	**600**	**1200**
RoamingOutgoingSTDRsSEC	**500.1**	**999.9**	**999.9**	**999.9**	**300**	**600**
Monthly Rental	**999**	**99**	**549**	**149**	**1499**	**99**
Total Monthly Bill	**₹ 16,537.60**	**₹ 24,363.00**	**₹ 20,212.96**	**₹ 28,336.90**	**₹ 14,705.50**	**₹ 22,658.50**

Note:

ALL: Instead of a numeric value, ALL means the entire usage for that line item is discounted. E.g. if DiscountSMSLocalNUM is ALL, then the cost incurred for sending Local SMSs is zero for the consumer

Cheapest Plan: If two or more plans have the same value and are cheapest for the given usage set, then all those plans must be submitted

UsageSet	Name of the usage set
OnNetLocalSEC	Seconds Anu spent in local OnNet outgoing calls in the month 201209
OffNetLocalSEC	Seconds Anu spent in local OffNet outgoing calls in the month 201209
OnNetSTDSEC	Seconds Anu spent in STD OnNet outgoing calls in the month 201209
OffNetSTDSEC	Seconds Anu spent in STD OffNet outgoing calls in the month 201209
ISD1SEC	Seconds Anu spent in ISD Group One outgoing calls in the month 201209
ISD2SEC	Seconds Anu spent in ISD Group Two outgoing calls in the month 201209
SMSLocalSMS	Number of local SMSs sent by Anu in the month 201209
SMSSTDSMS	Number of STD SMSs sent by Anu in the month 201209
SMSISDSMS	Number of ISD SMSs sent by Anu in the month 201209
SMSRoamingSMS	Number of SMSs sent by Anu when in roaming in the month 201209
RoamingIncomingSEC	Seconds Anu spent receiving calls when in roaming in the month 201209
RoamingOutgoingLocalSEC	Seconds Anu spent in outgoing local calls when in roaming in the month 201209
RoamingOutgoingSTDSEC	Seconds Anu spent in outgoing STD calls when in roaming in the month 201209

PlanName	Name of the plan
MonthlyRentalRs	Monthly rental (must be included in all calculations)
DiscountOnNetLocalSEC	The number of seconds from (OnNetLocalSEC) that must be discounted as free
DiscountOffNetLocalSEC	The number of seconds from (OffNetLocalSEC) that must be discounted as free
DiscountOnNetSTDSEC	The number of seconds from (OnNetSTDSEC) that must be discounted as free
DiscountNationalSec	The number of seconds from (OnNetLocalSEC + OffNetLocalSEC + OnNetSTDSEC + OffNetSTDSEC) that must be discounted as free. If (OnNetLocalSEC+OffNetLocalSEC+OnNetSTDSEC+OffNetSTDSEC) > DiscountNationalSec, then the lowest rate of (OnNetLocalSEC, OffNetLocalSEC, OnNetSTDSEC, OffNetSTDSEC) must be discounted first, followed by second lowest etc
DiscountSMSLocalNUM	The number of SMSs from (SMSLocalSMS) that must be discounted as free
DiscountSMSSTDNUM	The number of SMSs from (SMSSTDSMS) that must be discounted as free
DiscountSMSNationalNUM	The number of SMSs from (DiscountSMSLocalNUM + DiscountSMSSTDNUM) that must be discounted as free. If (DiscountSMSLocalNUM + DiscountSMSSTDNUM) > DiscountSMSNationalNUM, then the lowest rate of (DiscountSMSLocalNUM, DiscountSMSSTDNUM) must be discounted first, followed by second lowest
OnNetLocalRsSEC	Multiply this with OnNetLocalSEC to get the cost for this line item in rupees
OffNetLocalRsSEC	Multiply this with OffNetLocalSEC to get the cost for this line item in rupees
OnNetSTDRsSEC	Multiply this with OnNetSTDSEC to get the cost for this line item in rupees
OffNetSTDRsSEC	Multiply this with OffNetSTDSEC to get the cost for this line item in rupees
ISD1RsSEC	Multiply this with ISD1SEC to get the cost for this line item in rupees
ISD2RsSEC	Multiply this with ISD2SEC to get the cost for this line item in rupees
SMSLocalRsSMS	Multiply this with SMSLocalSMS to get the cost for this line item in rupees
SMSSTDRsSMS	Multiply this with SMSSTDSMS to get the cost for this line item in rupees
SMSISDRsSMS	Multiply this with SMSISDSMS to get the cost for this line item in rupees
SMSRoamingRsSMS	Multiply this with SMSRoamingSMS to get the cost for this line item in rupees
RoamingIncomingRsSEC	Multiply this with RoamingIncomingSEC to get the cost for this line item in rupees
RoamingOutgoingLocalRsSEC	Multiply this with RoamingOutgoingLocalSEC to get the cost for this line item in rupees
RoamingOutgoingSTDRsSEC	Multiply this with RoamingOutgoingSTDSEC to get the cost for this line item in rupees