Cellphone Broker

Cellphone plans are getting out of control!! There are some many components to people’s plans, voice minutes, data minutes, long distance, roaming etc… It’s becoming increasingly difficult to determine which plan is best.

Won’t it be great if someone would act as a broker (as they do in other industries) who could just analyze your calling patterns and determine the best plan for you (or even charge you based on it).

You have been approached by a company who wants to do this as a business and you have been asked to implement a program to make this possible. The owner of the company is a big OOP fan and demands that everything is fully implemented as objects for the maximum maintenance and expandability.

In this assignment we are particularly looking to demonstrate inheritance, method overloading, overriding and polymorphism in real situations to show the client our expertise in OOP.

The cellphone system must do all of the following (Scope Document):

* Allow for all the different aspects of cellphone charges
  + For each of Voice and Data
    - Minute charges
    - Minute limits
  + Long distance charges (we will assume no roaming)
  + Monthly plan charges
* Plans for each of the major cell phone companies
* Track a set of clients given their calling patterns
* Monthly routines which will:
  + Determine which plan to match the clients pattern
  + Report on the cost to the client
* Produce reports detailing
  + Total costs owing to each of the cellphone companies for their services.

Ultimately you will be presenting your solution to the class as a sales pitch to get the business. You need to take the appropriate time to make your presentation as slick as possible if you want to win the contract. Focus your presentation on the high level of the design although prototype demonstrations are certainly encouraged.

In an added twist, each person in the group must speak to one of each aspect of OOP in your solution:

1) Objects and Classes (overview)

2) Properties and Methods

3) Inheritance

4) Overloading

5) Overriding

6) Polymorphism

Which person speaks to each topic will be drawn at random before the presentation which means \*all\* members must be familiar with the whole presentation. This is common business practice in case someone is suddenly unavailable or your team must split up to present to multiple groups.

**Marking Scheme**

|  |  |  |
| --- | --- | --- |
| **Category** | **For What** | **Marks** |
| Think | Design show effective use of Classes, Objects, Properties and Methods | 4 |
| Think | Design show effective use of Inheritance, overloading, overriding and polymorphism | 4 |
| App | Implementation of prototype solution | 4 |
| App | Prototype demonstration targets customer requirements | 4 |
| Comm | Presentation was effective and had clear understanding on OOP concept | 4 |
| Comm | Presentation was creative and an effective pitch to get business from client | 4 |
|  | Total | /24 |