**Assignment #4 - Pay and Go Evening Parking**

**Pairs Programming Assignment**You may work with a partner on this assignment as long as you notify your instructor BEFORE you submit your design work. You can NOT designate a partner after the deadline for submitting the design. Submit only ONE copy of the assignment under one student account. Make sure BOTH names are included in the documentation AT THE TOP of the source code.

**Description**

You have recently been employed at Pay and Go Parking by their IT department as a Junior Developer. They had started work on a program to keep track of cars parked in your parking lot. The programmer has started work on the parking system but didn’t get a chance to finish the code.

This parking system is a bit different in that drivers don’t need to put a ticket on their dash, all they need to do is provide their licence plate number in conjunction with a credit card number.

This is an evening only parking lot and cars are charged a flat rate of $4.00 for the evening. Customers will come up to the machine and enter their credit card number and their license plate and then they can park in the lot.

There will be secret codes that will perform maintenance functions that only the parking attendant will know. The following is the list of the special functions:

* 22351 - plateList(): the program will output a list of the licence plates for all cars that have paid to park in the lot to the file ‘c:/temp/registered.txt’. The attendant will then go through the parking lot to determine which users have paid and which have not Extra marks will be given for providing a sorted list.
* 22352 – printCharges() will print a list of all the charges for the day to the file ‘c:/temp/charges.txt’
* 22353 – clearCars() – will remove all cars from the lot in preparation for a new set of customers tomorrow
* 22359 – used to shut down the program

The solution will require an array or an arraylist to track all the licence plates and credit card numbers. **Remember the purpose of this assignment is to verify that you know how to create methods and pass parameters to methods, so to get full marks for this assignment you must demonstrate that you know how to do this.**

Bonus:

* A bonus will be given for sorting the plate list and credit card list when output to the file.

**PROVIDE THE EXACT SAMPLE RUN SHOWN BELOW**, user input is shown in **bold underline**:

Welcome to Park and Go Parking

Park from 6 - Midnight for a flat fee of $4.00

1. Register your vehicle

2. Verify vehicle registration

Hello enter a Selection: **1**

Register your vehicle

Enter your plate number: **DEF127**

Enter your Credit Card Number ($4.00 charge): 5191-2301-2033-1289

Thank you, your plate DEF127 has been added to the lot.

Please enter to continue...

Welcome to Park and Go Parking

Park from 6 - Midnight for a flat fee of $4.00

1. Register your vehicle

2. Verify vehicle registration

Hello enter a Selection: **1**

Register your vehicle

Enter your plate number: **ABC465**

Enter your Credit Card Number ($4.00 charge): 5191-2301-1293-9233

Thank you, your plate ABC465 has been added to the lot.

Please enter to continue...

Welcome to Park and Go Parking

Park from 6 - Midnight for a flat fee of $4.00

1. Register your vehicle

2. Verify vehicle registration

Hello enter a Selection: **2**

Verify your registration

Enter your plate number: **ABC465**

You are registered in the lot

Please enter to continue...

Welcome to Park and Go Parking

Park from 6 - Midnight for a flat fee of $4.00

1. Register your vehicle

2. Verify vehicle registration

Hello enter a Selection: **2**

Verify your registration

Enter your plate number: **KHU192**

You are NOT registered,

please register and pay the $4.00 flat fee.

Please enter to continue...

Welcome to Park and Go Parking

Park from 6 - Midnight for a flat fee of $4.00

1. Register your vehicle

2. Verify vehicle registration

Hello enter a Selection: **22351**

Welcome to Park and Go Parking

Park from 6 - Midnight for a flat fee of $4.00

1. Register your vehicle

2. Verify vehicle registration

Hello enter a Selection: **22352**

Welcome to Park and Go Parking

Park from 6 - Midnight for a flat fee of $4.00

1. Register your vehicle

2. Verify vehicle registration

Hello enter a Selection: **22353**

Welcome to Park and Go Parking

Park from 6 - Midnight for a flat fee of $4.00

1. Register your vehicle

2. Verify vehicle registration

Hello enter a Selection: **2**

Verify your registration

Enter your plate number: **ABC465**

You are NOT registered,

please register and pay the $4.00 flat fee.

Please enter to continue...

Welcome to Park and Go Parking

Park from 6 - Midnight for a flat fee of $4.00

1. Register your vehicle

2. Verify vehicle registration

Hello enter a Selection: **22359**

Resultant data files:

charges.txt (unsorted, no bonus)

Daily parking summary for Fri Nov 15 09:20:33 MST 2013

Plate Credit Card Charge

================================================

DEF127 5191-2301-2033-1289 $ 4.00

ABC465 5191-2301-1293-9233 $ 4.00

================================================

Total $ 8.00

registered.txt (unsorted, no bonus)

Plate List for Fri Nov 15 09:20:25 MST 2013

Plate

===============

DEF127

ABC465

Here is the code that he has already written:

import java.io.\*;

import java.util.\*;

public class ass4ParkingStart {

public static void main (String [] args) throws IOException {

Scanner k = new Scanner (System.in);

int numCars = 0;

int s = 0;

boolean found;

s = printMenu(k);

k.nextLine();

while (s != 22359) {

switch (s)

{

case 1:

// Add a car to the lot

System.out.println ("Add a car here!");

System.out.print ("Please enter to continue...");

k.nextLine();

break;

case 2:

// verify plate

System.out.println ("Verify a plate here!");

System.out.print ("Please enter to continue...");

k.nextLine();

break;

case 22351:

// produce a report of all the plates

System.out.println ("Produce a report of all plates to the file” +

“ registered.txt here!");

System.out.print ("Please enter to continue...");

k.nextLine();

break;

case 22352:

// Produce a summary of charges

System.out.println ("Write a report of all charges to charges.txt here");

System.out.print ("Please enter to continue...");

k.nextLine();

break;

case 22353:

// clear out all the cars from the parking lot

System.out.println ("Remove all cars from the list here!");

System.out.print ("Please enter to continue...");

k.nextLine();

break;

default:

}

s = printMenu(k);

k.nextLine();

}

}

static public int printMenu(Scanner k) {

int s = 0;

// Clear the screen in BlueJ

System.out.print ('\u000C'); // comment this out BEFORE providing your sample run

System.out.println ("Welcome to Park and Go Parking");

System.out.println ("Park from 6 - Midnight for a flat fee of $4.00");

System.out.println ("1. Register your vehicle");

System.out.println ("2. Verify vehicle registration");

System.out.print ("Hello enter a Selection: ");

s = k.nextInt();

return s;

}

// Write all your functions here!

}

It is required that you subdivide the program into the functions to solve the:

1. Name: addCars
   * Purpose: Function: Prompts the user for their license plate number and credit card number and places that information at the end of the two arrays. Returns the new number of cars in the lot (e.g. numCars incremented by one).
   * Arguments
     + plates - is a list of all license plates of people that have paid to park in the lot
     + ccNumbers - is list of credit card numbers that are to be charged the flat $4 fee for evening parking
     + numCars (optional – depending on your implementation) – is the num of cars that have paid to park
     + k - a Scanner object attached to the keyboard in the main function

Worth 20% of mark

1. Name: plateList
   * Purpose: Print a report of license plates of all registered vehicles to the file c:/temp/registered.txt. User enters secret code 22351 to invoke this function. Since this function does file I/O make sure you throws IOException.
   * Arguments:
     + plates - is an list of all license plates of people that have paid to park in the lot
     + numCars (optional – depending on your implementation) – is the num of cars that have paid to park

Worth 30%

1. Name: printCharges

* Function: Print a report of license plates and credit card numbers that need to be manually charged for parking on a specific evening to the file dailyCharges.txt. User enters secret code 22352 to invoke this function. Since this function does file I/O make sure you throws IOException.
* Arguments:
  + plates - is an list of all license plates of people that have paid to park in the lot
  + ccNumbers - is list of credit card numbers that are to be charged the flat $4 fee for evening parking
  + numCars (optional – depending on your implementation) – is the num of cars that have paid to park
  + plates – is an array of all license plates of people that have paid to park in the lot

Worth 20% of mark

1. Name: verifyPlate
   * Function: Allows the user to enter a plate number and then searches to see if the plate is found the array ‘plates’
   * Arguments:
     + plates - is a list of all license plates of people that have paid to park in the lot
     + numCars (optional – depending on your implementation) – is the num of cars that have paid to park
     + k - a Scanner object attached to the keyboard in the main function

Worth 20% of mark

1. Name: clearCars
   * Function: Clears out all the elements in the plates and ccNumbers lists. User enters secret code 22353 to invoke this function.
   * Arguments:
     + plates - is an list of all license plates of people that have paid to park in the lot
     + ccNumbers - is list of credit card numbers that are to be charged the flat $4 fee for evening parking
     + numCars (optional – depending on your implementation) – is the num of cars that have paid to park

Worth 20%

Hints: Make sure that you have the files “registered.txt” and “charges.txt” closed (not open in notepad) when you run this program; otherwise, your program will not be able to write to these files. Before **producing your sample run comment out the line:**

**System.out.print ('\u000C');**

**in printMenu() so the screen is NOT cleared after each entry.** Turn on unlimited buffering in BlueJay to capture the entire session in the terminal window Terminal Windows->Options->Unlimited Buffering. To receive full marks on this assignment you MUST demonstrate the ability to use functions and pass parameters. If you choose to subdivide the program differently than the design as proposed in the structure that is acceptable.

**Submissions**  
Be careful to submit your own work, I will be using various tools to analyse all the submitted assignments to determine if any plagiarism has occurred. It is better to submit your own partial solution than to copy from another student.

It is recommended that you develop this program incrementally so that you can be awarded part marks for a working program, even if not all of the functions are working. Recommended order:

1. main function and printMenu()
2. addCars() – to add cars to the list
3. plateList() – to produce a list of cars to the file
4. verifyPlate() – confirms if the plate is registered in the parking lot
5. printCharges() - list of daily charges for parking in the lot
6. clearCharges() – removes all cars for the parking lot

Marking Rubric:

* Working Java code (required to receive a grade > 50%)
* Functionality supported:
  + addCars() – 20%
  + plateList() – 30%
  + verifyPlate() – 20%
  + printCharges() – 20%
  + clearCharges() – 10%
* Indentation – consistent
* Readability – good variable names
* Documentation
* Comments at the top which include: Name, date, program description
* Sample run that verifies the results against what was submitted in the test plan
* Bonus:
  + 10% sorted plate list and credit card list.