Design Document and Test Plan

Name of team members who collaborated on the design and test plan:

1. Name (*first last*): Ben Curry
2. Name (*first last*): Reeves Farrell
3. Name (*first last*): Alex Gravlee
4. Name (*first last*): Jordyn Brooks

Name of programming for which you submit this document: **Parking Permits**

UML Class Diagrams

Classes Needed (4 of each)

- CustomerInfo
  - Student, Employee, Visitor, Service (ie. construction workers)
  - name, address, email, and at least two class-specific attributes
- VehicleInfo
  - LEV, Car, Motorcycle, Moped
  - make, model, year, license plate #, and at least two class-specific attributes
- Invoice
  - stores the price for the selected permit, at least one type of discount that you decide and at least one service charge you decide
  - Member functions
    - one calcs total price - one outputs full invoice to screen
- Permit type
  - annual, semester, one-day, park-and-ride
- .txt files
  - inputTest1.txt - all input for program
  - outputTest1.txt - expected output of the program based on inputTest1.txt inputs
  - inputTest2.txt/outputTest2.txt
  - inputTest3.txt/outputTest3.txt

Customer Type Classes:

Jordyn - Student
- name: string - address: string - email: string - studentID: string - classStanding: int
+ getName(): string + getAddress(): string + getEmail(): string + getStudentID(): string + getClassStanding(): int + setName(n: string): bool + setAddress(a: string): bool + setEmail(e: string): bool + setStudentID(s: string): bool + setClassStanding(y: int): bool

Alex - Service
- name : string - address : string - email : string - department (i.e. maintenance) : string - yearsService: int
+ getName(): string + getAddress(): string + getEmail() : string + get department(): string + get yearsService(): int + setName(): bool + setAddress(): bool + setEmail() : bool + set department(): bool + set yearsService(): bool

Reeves - Visitor
- name: string - address: string - email: string - duration: int - parent: string
+ getName(): string + getAddress(): string + getEmail(): string + getDuration(): int + getParent(): string + setName(n: string): bool + setAddress(a: string): bool + setEmail(e: string): bool + setDuration(d: int): bool + setParent(p: string): bool

Ben - Employee
- name: str - address: str - email: str - employeeIDNum: int - employeeJob: str
+ setName(n: string): bool + setAddress(add: string): bool + setEmail(e: string): bool + setEmployeeID(ei: int): bool + setEmployeeJob(ej: str): bool + getName(): str + getAddress(): str + getEmail(): str + getEmployeeID(): int + getEmployeeJob(): str

Vehicle class types:

Jordyn - LEV
- make: string - model: string - year: int - licensePlate: string - chargeTime: double - gallons: double
+ getMake(): string + getModel(): string + getYear(): int + getLicensePlate(): string + getChargeTime(): double + getGallons(): double + setMake(m: string): bool + setModel(o: string): bool + setYear(y: int): bool + setLicensePlate(l: string): bool + setChargeTime(t: double): bool + setGallons(g: double): bool

Alex - Moped
- make: string - model: string - year: int - licensePlate: string - mopedPower(CCs): int - commuteDist(nearest mile): int
+ getMake(): string + getModel(): string + getYear(): int + getLicensePlate(): string

+ getMopedPower(): int + getCommuteDist(): int + setMake(): bool + setModel(): bool + setYear(): bool + setLicensePlate(): bool + setMopedPower(): bool + setCommuteDist(): bool

Ben - Car
- make: str - model: str - year: int - licensePlateNum: str - gasType: str - mpg: int
+ setMake(ma: string): bool + setModel(mo: string): bool + setYear(y: int): bool + setLPN(lpn: string): bool + setGasType(gt: string): bool + setMPG(MPG: int): bool + getMake(): str + getModel(): str + getYear(): int + getLPN(): str + getGasType(): str + getMPG(): int

Reeves - Motorcycle
- make: string - model: string - year: int - licensePlate: string - wheels: int

- sideCar: bool
+ getMake(): string + getModel(): string + getYear(): int + getLicensePlate(): string + getWheels(): int + getSideCar(): bool + setMake(ma: string): bool + setModel(mo: string): bool + setYear(y: int): bool + setLicensePlate(lp: string): bool + setWheels(w: int): bool + setSideCar(sc: bool): bool

Invoice
- price: double - discount: double - serviceCharge: const int
+ calcTotalPrice(): int + printInvoice(): string ■ printInvoice(), getMake(), getModel(), etc.)

Pseudocode

(See Ch. 1.6 in our textbook for an example of how to write detailed pseudocode)

Ask the user if they are a Student, Employee, Visitor, or Service worker. Store user input.

Ask for and store user type

Depending on the users answer, instantiate needed customer info class object and invoice class object

Collect necessary customer information depending on type of pass being purchased

Ask user which type of vehicle they wish to purchase a pass for

Instantiate correct vehicle class object needed

Prompt for, and store, required information

Calculate total charge based on collected information

Output invoice

Costs:

Cars:

Student = $173

Employee = $150

Visitor = $30

Service = $150

Moped: $75

Motorcycle: $75

LEV: same as car price plus discount (20%)

Test Plan

(See Ch. 5.13 in our textbook for an example of how to write a test plan)

Test #	Purpose	Input	Expected Output
1	Check Employee class and Car Class	Employee, Ben Curry, 2098 Woodcrest Circle, professor@school.edu, Professor, 123456, Car, Subaru, Outback, 2010, 87 Octane, 123456, 12.1	Pass Purchased: Employee Name: Ben Curry Address: 2098 Woodcrest Circle Email: professor@school.edu Job Title: Professor Employee ID: 123456 Vehicle Type: Car Vehicle make: Subaru Vehicle model: Outback Year: 2010

			Gas type: 87 Octane License plate number: 123456 MPG: 12.1 Total: $150
2	Check Student Class and LEV Class	Student, Jordyn Brooks, 107 W Leroy St, jkbrook@clemson.edu, C14535869, 3, Toyota, Camry, 2006, PPG720, 40, 15	Pass Purchased: Student Name: Jordyn Brooks Address: 107 W Leroy St Email: jkbrook@clemson.edu Student ID: C14535869 Class Standing: Junior Make: Toyota Model: Camry Year: 2006 License Plate Number: PPG720 Charge Time: 40 minutes Tank Size: 15 gallons Total: $138.40
3	Check Service Class and Moped Class	Service, Alex Gravlee, 106 Crooked Cedar Way, agravle@clemson.edu, Facilities, 5, Italjet, Dragster, 2020, SBY782, 125, 8	Pass Purchased: Service Name: Alex Gravlee Address: 106 Crooked Cedar Way Email: agravle@clemson.edu Department: Facilities Years Service: 5 Make: Italjet Model: Dragster Year: 2020 License Plate: SBY782 Moped Power: 125 Commute Distance: 5
4	Check Visitor Class and Motorcycle Class	Visitor, Reeves Farrell, 111 Circle Street, jrfarre@clemson.edu, 1, Harley Davidson, Iron 883, 2007, AB178P, 2, No	Pass Purchased: Visitor Name: Reeves Farrell Address: 111 Circle Street Email: jrfarre@clemson.edu Duration: 24 hours Parent: Yes Make: Harley Davidson Model: Iron 883 Year: 2007 License Plate Number: AB178P Wheels: 2 Sidecar: No Total: $75
5	Check Student Class and Car Class	Student, Jordyn Brooks, 107 W Leroy St,	Pass Purchased: Student Name: Jordyn Brooks

		jkbrook@clemson.edu, C14535869, 3, Toyota, Camry, 2006, PPG720, 87 Octane, 28	Address: 107 W Leroy St Email: jkbrook@clemson.edu Student ID: C14535869 Class Standing: Junior Make: Toyota Model: Camry Year: 2006 License Plate Number: PPG720 Gas Type: 87 Octane MPG: 28 Total: $138.40
…	*(Feel free to add more test cases)*		