

	<b>Course:</b>	<b>OOP Lab</b>	<b>Code:</b>	<b>CL217</b>
	<b>Program:</b>	<b>BS (Computer Science)</b>	<b>Semester:</b>	<b>Spring 2019</b>
	<b>Duration:</b>	<b>130 minutes</b>	<b>T. Marks:</b>	<b>100</b>
	<b>Date:</b>	<b>Monday 29-04-2019</b>	<b>Weight</b>	<b>40%</b>
	<b>Section:</b>	<b>ALL</b>	<b>Page(s):</b>	<b>2</b>
	<b>Exam:</b>	<b>Lab FINAL Term</b>	<b>Roll No:</b>	

#### Instructions/Notes:

- It is your responsibility to make sure that **CACTUS** is accessible through your PC.
- **Keep backup** (save project in drive D) of your final-term's code files and update the files frequently.
- Use of the internet, notes, codes, lab manuals, and flash drives are strictly prohibited.
- Plagiarism will result in **F** grade in lab.
- Please **read the “Main() function”** before starting the code.
- Submission path: Section-X (here X will be your section A or B or C ...)  
<\\cactus\xeon\Spring 2019\Shakeel Zafar\Final OOP\Section-X>
- Code must be **indented properly**, failure to comply will incur a penalty.

#### Question # 1:

You need to implement an amateur level system for a device and its applications. A **Device** can be of two types i.e **Mobile** and **Desktop**. There are two types of **Application** i.e **Calls** and **Store**. A Device may have zero or many applications. An Application from above mentioned types can be installed in Device and can be uninstalled as well. After installation, application can be run to execute its functionalities. Functionalities of specific applications are given below.

You should implement (constructors, destructors & functions) for following classes:

##### 1) Class **Application** (Parent class)

With following members & functions

- It will have private variable  
**Name** (char \*), **Version** (float)
- It will have a function **print**  
To display all the features of the Application.
- It has the function **run**  
Which is pure virtual function.

##### 2) Class **Store** (Derived class from Application)

With following members & functions

- For the time being this class has no data member.
- It will have a function **print**  
To display all the features of the Application (Think about, how you can use the function in parent class).
- It will have a function **run** (overrides the function from parent class)  
Which displays some dummy new applications (Upon your wishes) and returns.

##### 3) Class **Calls** (Derived class from Application)

With following members & functions

- It will have a private variable  
**totalCalls** (int\*)
- It will have a function **print**  
To display all the features of the Calls. (Think about, how you can use the function in parent class).
- It will have a function **run** (overrides the function from parent class)  
Which displays some dummy new Calls and asks user to make a new call.  
You have to make a function **makeNewCall**. This function executes a cout statement i.e “New Call, Done”. And increments the variable totalCalls.

#### 4) Class **Device** (Parent class)

With following members & functions:

- It will have protected dynamic variables for  
**listOfApplications** (Application \*\*), **totalApplications** (Int)  
**RAM** (Int), **Processor** (float), **OS Version** (char \*),
- It will have a function **print ()**  
To display all the features of Device.
- It will have a function **onStart()**  
Which displays a menu to user
  1. List all applications (prints all installed apps on screen)
  2. Install an application (makes a new object and insert in list)
  3. Uninstall an application (deletes an object from the installed list)
  4. Run an application (asks user which application you want to run and calls the run function of that application)

You have to implement separate functions for all the above menu options.

#### 5) Class **Mobile** (Derived class from Device)

With following functions

- It will have private variable  
**Camera** (Int)
- It will have a function **print**  
To display all the features of the Mobile. (Think about, how you can use the function in parent class).

#### 6. Class **Desktop** (Derived class from Device)

With following functions

- It will have private variable  
**gpu** (Boolean)
- It will **printinfo**.  
To display all features of **Desktop**.

#### **main ()** function

This function will offer following functionalities:

- Create two pointers of type **Device** and which will hold two objects of **Mobile** and **Desktop**.
- With the construction of object of these Mobile/Desktop classes, you have to send data of required data members. Initially Desktop/Mobile class objects will have no applications installed.
- Call the function **onStart()**, install at least two applications in every device. **Installation Details:** You have to create an object of any type of application and that object will be pointed by a pointer from **listOfApplications**. You have to maintain the size of these applications. This **listOfApplications** will dynamically increase and decrease according to the applications installed in this device.
- User must be allowed to run at least an application in each device.
- User must be allowed to uninstall any installed application.  
**Uninstallation Details:** You have to uninstall an application. Which means you have to delete an object from **listOfApplications**, so you have shrink the memory.
- You have to print all the devices details and application details on screen for all applications.

Since you are using dynamic variables, **you need to manage memory effectively and make sure that proper destructors** are called.

-----**BEST of LUCK**-----