(a) Write the `Station` member function `ResetAll`, as started below. `ResetAll` changes the number of gallons sold at each pump to 0.0.

In writing `ResetAll`, you may call any of the public member functions of the `Pump` and `Station` classes. Assume that all these functions work as specified.

Complete function `ResetAll` below.

```cpp
void Station::ResetAll()
{ // postcondition: For every Pump p in this station
    // p.GallonsSold() is 0.0
    for(int x = 0; x < myPumps.length(); x++)
        myPumps[x].ResetGallonsSold();
}
```
(b) Write the Station member function TotalSales, as started below. TotalSales returns the total cash value of the gallons sold at all pumps. Recall that self-service pumps charge the base price for each gallon of gas, while full-service pumps (pumps 0 and 1) charge $0.25 more per gallon.

In writing TotalSales, you may call any of the public member functions of the Pump and Station classes. Assume that all these functions work as specified.

Complete function TotalSales below.

```cpp
double Station::TotalSales() const
    // postcondition: returns the total cash value of sales
    // for all pumps
{
    double sum = 0.0;
    double x = myBasePrice + 0.25;
    for (int a = 0; a < 2; a++)
        sum += (x * myPumps[a].GallonsSold());
    for (int k = 2; k < myPumps.length(); k++)
        sum += (myBasePrice * myPumps[k].GallonsSold());
    return sum;
}
```
(c) Write the `Station` member function `CloseStation`, as started below. `CloseStation` will write the total amount of money earned from the day's gas sales to the output stream, `logFile`, and then reset the number of gallons sold at each individual pump to 0.0.

In writing `CloseStation`, you may call any of the public member functions of the `Pump` and `Station` classes. Assume that these functions, including `ResetAll` and `TotalSales`, work as specified, regardless of what you wrote in parts (a) and (b).

Complete function `CloseStation` below.

```cpp
void Station::CloseStation(ostream & logFile)
{
    // precondition: logFile is open and ready for writing
    // postcondition: writes the total cash value of
    //                 all gas sold for the day to logFile;
    //                 for every Pump p in this station
    //                 p.GallonsSold() is 0.0

    logFile << TotalSales();
    ResetAll();
}
```
(a) Write the `Station` member function `ResetAll`, as started below. `ResetAll` changes the number of gallons sold at each pump to 0.0.

In writing `ResetAll`, you may call any of the public member functions of the `Pump` and `Station` classes. Assume that all these functions work as specified.

Complete function `ResetAll` below.

```cpp
void Station::ResetAll()
// postcondition: For every Pump p in this station
// p.GallonsSold() is 0.0
{
    for (int k = 0; k < myPumps.length; k++)
        p[k].GallonsSold() = 0.0;
```
(b) Write the Station member function TotalSales, as started below. TotalSales returns the total cash value of the gallons sold at all pumps. Recall that self-service pumps charge the base price for each gallon of gas, while full-service pumps (pumps 0 and 1) charge $0.25 more per gallon.

In writing TotalSales, you may call any of the public member functions of the Pump and Station classes. Assume that all these functions work as specified.

Complete function TotalSales below.

double Station::TotalSales() const
// postcondition: returns the total cash value of sales
// for all pumps
{
    double total, fulltot, selftot;
    Pump p;
    int k;
    for (k = 0; k < myPumps.length(); k++)
    {
        if ((p[k] == p[0]) || (p[k] == p[1]))
        {
            myBasePrice += .25;
            fulltot = myBasePrice * p[k].GallonsSold();
        }
        else
        {
            selftot = myBasePrice * p[k].GallonsSold();
        }
    }
    total = selftot + fulltot;
    return total;
}
(c) Write the \texttt{Station} member function \texttt{CloseStation}, as started below. \texttt{CloseStation} will write the total amount of money earned from the day's gas sales to the output stream, \texttt{logFile}, and then reset the number of gallons sold at each individual pump to 0.0.

In writing \texttt{CloseStation}, you may call any of the public member functions of the \texttt{Pump} and \texttt{Station} classes. Assume that these functions, including \texttt{ResetAll} and \texttt{TotalSales}, work as specified, regardless of what you wrote in parts (a) and (b).

Complete function \texttt{CloseStation} below.

```cpp
void Station::CloseStation(ostream & logFile)
// precondition: logFile is open and ready for writing
// postcondition: writes the total cash value of
// all gas sold for the day to logFile;
// for every Pump p in this station
// p.GallonsSold() is 0.0
{
    logFile << TotalSales( );
    ostream >> logFile;
    ResetAll( );
}
```
(a) Write the Station member function ResetAll, as started below. ResetAll changes the number of gallons sold at each pump to 0.0.

In writing ResetAll, you may call any of the public member functions of the Pump and Station classes. Assume that all these functions work as specified.

Complete function ResetAll below.

```c++
void Station::ResetAll()
// postcondition: For every Pump p in this station
// p.GallonsSold() is 0.0
{
    Pump p;
    int h;
    for (h = 0; h < p.Length(); h++)
    {
        p[h].GallonsSold() = 0.0;
    }
```
(b) Write the `Station` member function `TotalSales`, as started below. `TotalSales` returns the total cash value of the gallons sold at all pumps. Recall that self-service pumps charge the base price for each gallon of gas, while full-service pumps (pumps 0 and 1) charge $0.25 more per gallon.

In writing `TotalSales`, you may call any of the public member functions of the `Pump` and `Station` classes. Assume that all these functions work as specified.

Complete function `TotalSales` below.

double Station::TotalSales() const
// postcondition: returns the total cash value of sales
// for all pumps
{
  for (Pump pi : pumps)
    if (pi == p0 || pi == p1)  
      return (myBasePrice + .25);  
    else
      return myBasePrice;
}  

}
(c) Write the Station member function CloseStation, as started below. CloseStation will write
the total amount of money earned from the day's gas sales to the output stream, logFile, and then reset
the number of gallons sold at each individual pump to 0.0.

In writing CloseStation, you may call any of the public member functions of the Pump and
Station classes. Assume that these functions, including ResetAll and TotalSales, work as
specified, regardless of what you wrote in parts (a) and (b).

Complete function CloseStation below.

```cpp
void Station::CloseStation(ostream & logFile)
// precondition:  logFile is open and ready for writing
// postcondition:  writes the total cash value of
//                 all gas sold for the day to logFile;
//                 for every Pump p in this station
//                 p.GallonsSold() is 0.0

Pump *p;
ifstream in;
while (ResetAll())

    in >> TotalSales();
```

```