OOP244SCC Quiz 5 question bank; please print the first three pages and bring them with you to the class.
Story:

BarChart is a class designed to dynamically gather several sample integer values in itself and display them as a bar chart.

So a BarBhart holding the following values: 30, 50 and 40, will display itself as follows:

1:==============================
2:==
3:======================================

You can open **11-June20.vcxproj** and test the execution of the class. The source code of the BarChart is partially posted to help you understand how it works.

BarChart:

```
namespace sict {
   class BarChart {
      int* m_values;
      int m_size;
   public:
      BarChart(int size = 1);
      BarChart(const int vals[], int size);
      BarChart(const BarChart& BC);
      BarChart& operator=(const BarChart& BC);
      ~BarChart();
      std::ostream& display(std::ostream& ostr = std::cout)const;
      std::istream& read(std::istream& istr = std::cin);
      int max()const;
      int min()const;
      int average()const;
      BarChart& merge(const BarChart& BC);
   };
}
```

Code NOT provided (object file provided to be able to compile and run the program):

BarChart(int size = 1);
Dynamically creates a BarChart with an array of "size" integers to keep the samples. if size is not provided, an array of one sample will be allocated

BarChart(const int values[], int size);
Dynamically creates a BarChart with an array of "size" integers to keep the samples and sets them to the incoming values in the integer array “vals”

BarChart(const BarChart& BC);
BarChart& operator=(const BarChart& BC);
Copy constructor and assignment operator.

Code Provided:

std::ostream& display(std::ostream& ostr = std::cout)const;
Displays the sample values as series of bar charts at shown above, returning ostream

std::istream& read(std::istream& istr = std::cin);

Reads the values of samples from the keyboard one by one, returning istream;

`int max()const;`
returns the largest value in the samples

`int min()const;`
returns the smallest value in the samples

`int average()const;`
returns the average value of the samples

`BarChart& merge(const BarChart& BC);`
Merges a BarChart into the current one and returning he current object.

The following main program should be working as follows, assuming the user entering 10 and 20:

```
int main() {
   int vals[] = { 30,50,40 };
   BarChart B(2), C(vals, 3), A(C);
   cout << "Enter 2 Barchart value samples:" << endl;
   B.read();
   cout << "A: " << endl;
   A.display() << endl;
   cout << "B: " << endl;
   B.display() << endl;
   cout << "C: " << endl;
   C.display() << endl;
   cout << "Min value in B: " << B.min() << endl;
   cout << "Max value in C: " << C.max() << endl;
   cout << "B.merge(C) " << endl;
   A = B.merge(C);
   cout << "B: " << endl;
   B.display() << endl;
   cout << "C += B: " << endl;
   cout << "A: " << endl;
   A.display() << endl;
   cout << "Average value of all samples: " << C.average() << endl;
   return 0;
}
```

```
Enter 2 Barchart value samples:

1/2: 10

2/2: 20

A:

1:==============================

2:==================================================

3:========================================
```

B:

1:==========

2:====================

C:

1:==============================

2:==

3:==

Min value in B: 10

Max value in C: 50

B.merge(C)

B:

1:==========

2:====================

3:==============================

4:==

5:==

C += B:

A:

1:==========

2:====================

3:==============================

4:==

5:==

The Question bank: (some of these will be in the quiz)

1- Write the following constructor:
BarChart(int size = 1);
Dynamically creates a BarChart with an array of "size" integers to keep the samples. if size is not provided, an array of one sample will be allocated

2- Write the following constructor
BarChart(const int values[], int size);
Dynamically creates a BarChart with an array of "size" integers to keep the samples and sets them to the incoming values in the integer array “vals”

3- Write the Copy Constructor:

4- Write the assignment operator to set a BarChart to another;
```
BarChart& operator=(const BarChart& BC);
```

5- Write the descructor.

The following questions must be written ONLY by reusing the already existing methods:

6- Overload the unary + operator:
returns the largest value in the samples

7- Overload the unary – operator:
returns the smallest value in the samples

8- overload the integer cast operator to:
return the average value of the samples

9- Overload the binary += member operator to merge the right BarChart into the left one.
Merges a BarChart into the current one and returning he current object.

10- Overload helper operator >> to read the BarChart values using cin.

11- Overload helper operator << to Display the BarChart as displayed in this document.

12- Overload helper operator + to merge two BarCharts into a new BarChart and return it without changing the two original ones.

Implementing the above question the main should be able to be written as follows to have a similar (but not identical) output as the previous main:

```cpp
int main() {
   int vals[] = { 30,50,40 };
   BarChart B(2), C(vals, 3), A(C);
   cout << "Enter 2 Barchart value samples:" << endl;
   cin >> B;
   cout << "A: " << endl;
   cout << A << endl;
   cout << "B: " << endl;
   cout << B << endl;
   cout << "C: " << endl;
   cout << C << endl;
   cout << "Min value in B: " << -B << endl;
   cout << "Max value in C: " << +C << endl;
   cout << "A = B + C: " << endl;
   A = B + C;
   cout << A << endl;
   cout << "C += B: " << endl;
   C += B;
   cout << C << endl;
   cout << "Average value of all samples: " << int(C) << endl;
   cout << "A = B: " << endl;
   A = B;
   cout << A << endl;
   return 0;
}
```

```
Enter 2 Barchart value samples:
1/2: 10
2/2: 20
A:
1:==============================
2:==================================================
3:========================================

B:
1:==========
2:====================

C:
1:==============================
2:==================================================
3:========================================

Min value in B: 10
Max value in C: 50
A = B + C:
1:==========
2:====================
3:==============================
4:==================================================
5:========================================

C += B:
```

1:==============================
2:==
3:=======================================
4:==========
5:====================

Average value of all samples: 30
A = B:
1:==========
2:====================