National University of Computer and Emerging Sciences

Lab Manual # 04
Object oriented programming

Course Instructor	Ms. Arooj Khalil
Lab Instructor(s)	Memoona Akbar Saleha Batool
Section(s)	BSE-2B1 BSE-2B2
Semester	Spring 2023

Department of Computer Science

FAST-NU, Lahore, Pakistan

TASK:
Implement a class called **BiggerInt**. The BiggerInt class will have two data members:

- int* big_int_; // Pointer to the int array that holds the big integer
- int int_length_; // Variable to store the length of the big integer

(While an integer is of 4 bytes in sizde with a range of -2,147,483,648 to 2,147,483,647. A big integer can store long integer numbers with no size limitation.)

You have to implement the following:

1. Write a default constructor and initialize big_int_ to nullptr.
    - BiggerInt();

2. Write an overloaded constructor and perform deep copy.
    - BiggerInt (const int * obj, int size);

3. Write a member function to make a deep copy of the big_int_ of the passed BiggerInt obj into the big_int_ of the object which called this function.
- void assign(const BiggerInt & obj);

4. Write a member function which will overload the above assign function and performs the same operations but the argument passed to this function is a pointer integer array.
- void assign(const int * big_int, int size);

5. Write a member function to append the big_int_ of the passed BiggerInt obj to the end of big_int_ of the object which called this function.
- void append(const BiggerInt & obj);

6. Write a member function which will overload the above append function and performs the same operations but the argument passed to this function is a pointer integer array.
- void append(const int* big_int, int size);

7. Write a member function to compare the big_int_ of BiggerInt obj with the big_int_ of the object which called this function. Return 0 for equal, 1 for less than and 2 for greater than.

- int compareTo(const BiggerInt & obj);

8. Write a member function which overloads the above compareTo function and performs the same operations but the argument passed to this function is a pointer integer array.
- int compareTo(const int* big_int, int size);

9. Write a member function to display the big_int_ on screen. If big_int_ is empty, print “No Value Assigned”.
- void display();

10. Write a destructor to deallocate any dynamically allocated memory.
- ~ BiggerInt();

11. Write a suitable main() function in the driver.cpp to test all the functions of the BiggerInt class.

Note:
- Deallocate all dynamically allocated memory.
- Make separate my_big_int.h, my_big_int.cpp and driver.cpp files.
- Do not use any string class built-in functions except for strlen(), if required.
- Follow all the code indentation, naming conventions and code commenting guidelines.