|  |
| --- |
| **Bangladesh University of Business & Technology (BUBT)** |
| **Department of Computer Science and Engineering** |
| **Assignment - 03: Spring 2023** |
| **Course Code: CSE 122** │**Course Title: Object Oriented Programming Language Lab** |
| **Intake: 50th, Program: B.Sc in CSE (Bi-Semester)** |
| **Marks – 10** |

|  |  |  |
| --- | --- | --- |
| **CO** | **Question** |  |
| **CO3** | **Demonstrate** a C++ code that creates a class called *Fraction*. The class Fraction has two attributes: *numerator and denominator.*  • In your *constructor (inyour\_\_init\_\_ method)*, verify(assert?) that the numerator and denominator passed in during initiation are both of type int. If you want to be thorough, also check to make sure that the denominator is not zero.  • Write a *.reduce() method* that will reduce a fraction to lowest terms.  • Override the Object class’s \_\_str\_\_ and \_\_repl\_\_ methods so that your objects will print out nicely. Remember that\_\_str\_\_ is more for humans; \_\_repl\_\_ is more for programmers. Ideally ,the\_\_repl\_\_ method will produce a string that you can run through the eval() function to clone the original fraction object.  • Override the + operator. In your code, this means that you will implement the special method \_\_add\_\_. The signature of the \_\_add\_\_ function will be def \_\_add\_\_(self, other): , and you’ll return a new Fraction with the result of the addition. Run your new Fraction through the *reduce() function* before returning. |  |