1. Viết chương trình nhập vào 1 kí tự đó in ra “This is an uppercase” nếu kí tự đó là chữ cái in hoa, in ra “This is an lowercase” nếu ký tự đó là chữ cái viết thường và in ra “This is not an alphabetic” nếu kí tự đó không phải chữ cái.

2. Viết chương trình nhập vào 3 điểm A(x, y), B(x, y) và C(x, y) trong hệ trục tọa độ Oxy sau đó kiểm tra xem 3 điểm vừa nhập tọa độ có thỏa mãn tạo thành tam giác hay không, nếu có thì in ra chu vi và diện tích tam giác ABC nếu không thỏa mãn thì in ra “Invalid”.

3. Viết chương trình nhập vào 3 số nguyên a, b, c là các số cho trước của phương trình bậc hai ax^2 + bx + c = 0 sau đó tìm nghiệm và in giá trị của nghiệm ra màn hình nếu phương trình vô nghiệm thì in ra “No solution” (Ràng buộc a phải khác 0).

4. Viết chương trình nhập vào 2 số nguyên a và b > 0 sau đó trả về dạng tối giản của phân số a/b.

5. Viết chương trình nhập vào 1 chuỗi ký tự, kiểm tra xem chuỗi có phải palindrome hay không nếu đúng thì in ra “This is a palindrome”, nếu không thì in ra “This is not a palindrome” (Hướng dẫn: palindrome là một chuỗi nếu đảo ngược cũng sẽ thu được chuỗi ban đầu).

6. Viết chương trình nhập vào chuỗi số sau đó in ra tổng của các chữ số có trong chuỗi vừa nhập.

7. Viết hàm có tên perfectNum() hàm này có 1 tham số là một số nguyên > 0. Chức năng là kiểm tra và trả về true nếu số đầu vào là một số hoàn thiện, còn lại false (Hướng dẫn: số hoàn thiện là số mà tổng các ước số của nó không bao gồm chính nó bằng chính nó).

8. Viết hàm có tên factorial() hàm này có 1 tham số là một số nguyên >= 0. Chức năng là trả về số đầu vào lấy giai thừa.

9. Viết hàm có tên fibonacci() hàm này có 1 tham số đó là n số nguyên >= 0. Chức năng là trả về số fibonacci thứ n (Hướng dẫn: số fibo thứ n bằng tổng của số fibo thứ n - 1 và số fibo thứ n - 2, trong đó số fibo thứ 0 có giá trị là 0 và số fibo thứ 1 có giá trị là 1).

10. Viết hàm có tên cutString() hàm này có 3 tham số trong đó một tham số đầu vào s là một chuỗi, tham số đầu vào m là một số nguyên >= 0 và tham số đầu vào n còn lại là một số nguyên >= 0. Chức năng là trả về chuỗi được cắt từ chỉ mục m đến chỉ mục n từ chuỗi s.

11. Viết hàm có tên GCD() hàm này có 1 tham số là một bộ số nguyên > 0. Chức năng là trả về ước chung lớn nhất của bộ số đầu vào.

12. Viết hàm có tên productOfSum() hàm này có 1 tham số là một chuỗi số dương > 0. Chức năng là ta sẽ tính tổng của tất cả các chữ số có trong chuỗi số đầu vào, sau đó ta tính tích của tất cả các chữ số có trong tổng vừa tính được ở trên, cuối cùng trả về kết quả của tích vừa tìm được.

13. Viết hàm có tên armstrongNumber() hàm này có 1 tham số là một số nguyên >= 0. Chức năng là kiểm tra xem tham số đầu vào có phải là armstrong hay không, nếu đúng thì trả về true, còn không trả về false (Hướng dẫn: số armstrong là số mà tổng lũy thừa số lượng chữ số của các chữ số trong nó bằng chính nó).

14. Viết hàm có tên sortArray() hàm này có 1 tham số là một mảng số nguyên. Chức năng là sắp xếp mảng đầu vào thêm thứ tự tăng dần về giá trị và hàm này sẽ không trả về bất kỳ giá trị nào.

15. Tìm hiểu và triển khai thuật toán tìm kiếm linear search.

16. Tìm hiểu và triển khai thuật toán sắp xếp selection sort.

17. Thực hiện các yêu cầu sau:

Tạo ra lớp Point:

- Trường:

+ int x: tọa độ x.

+ int y: tọa độ y.

- Phương thức:

+ Point() là constructor mặc định.

+ Point(int x, int y) là constructor khởi tạo 2 trường x và y từ 2 tham số đầu vào.

+ getX(), setX(int y), getY(), setY(int y) là các getter và setter.

+ toString() là phương thức trả về chuỗi có dạng:

“Point [x=tọa độ x, y=tọa độ y]”

Tạo ra lớp Triangle:

- Trường:

+ Point A: tọa độ điểm A.

+ Point B: tọa độ điểm B.

+ Point C: tọa độ điểm C.

- Phương thức:

+ Triangle() là constructor mặc định.

+ Triangle(Point A, Point B, Point C) là constructor khởi tạo 3 trường A, B, C từ 3 tham số đầu vào.

+ getA(), setA(Point A), getB(), setB(Point B), getC(), setC(Point C) là các getter và setter.

+ getAB(), getBC(), getAC() là các phương thức trả về độ dài các cạnh tương ứng AB, BC, AC.

+ isValid() là phương thức trả về true nếu tọa độ 3 điểm A, B, C thỏa mãn tạo thành tam giác, còn không trả về false.

+ perimeter() là phương thức trả về chu vi tam giác ABC nếu tọa độ 3 điểm A, B, C thỏa mãn tạo thành tam giác, còn không trả về 0.

+ area() là phương thức trả về diện tích tam giác ABC nếu tọa độ 3 điểm A, B, C thỏa mãn tạo thành tam giác, còn không trả về 0.

+ toString() là phương thức trả về chuỗi có dạng:

“Triangle [A=tọa độ điểm A, B=tọa độ điểm B, C=tọa độ điểm C][perimeter=chu vi tam giác, area=diện tích tam giác]”

18. Thực hiện các yêu cầu sau:

Tạo ra lớp Fraction:

- Trường:

+ int numer: tử số.

+ int denom: mẫu số.

- Phương thức:

+ Fraction() là constructor mặc định.

+ Fraction(int numer, int denom) là constructor khởi tạo 2 trường numer và denom từ 2 tham số đầu vào.

+ getNumer(), setNumer(int numer), getDenom(), setDenom(int denom) là các getter và setter.

+ toString() là phương thức trả về chuỗi có dạng:

“Fraction [tử số/mẫu số]”

Tạo ra lớp trừu tượng FractionMath:

- Phương thức trừu tượng:

+ add(Fraction f)

+ minus(Fraction f)

+ time(Fraction f)

+ div(Fraction f)

+ simplify(Fraction f)

Cho lớp Fraction kế thừa từ lớp trừu tượng FractionMath, sau đó triển khai các phương thức trừu tượng sau:

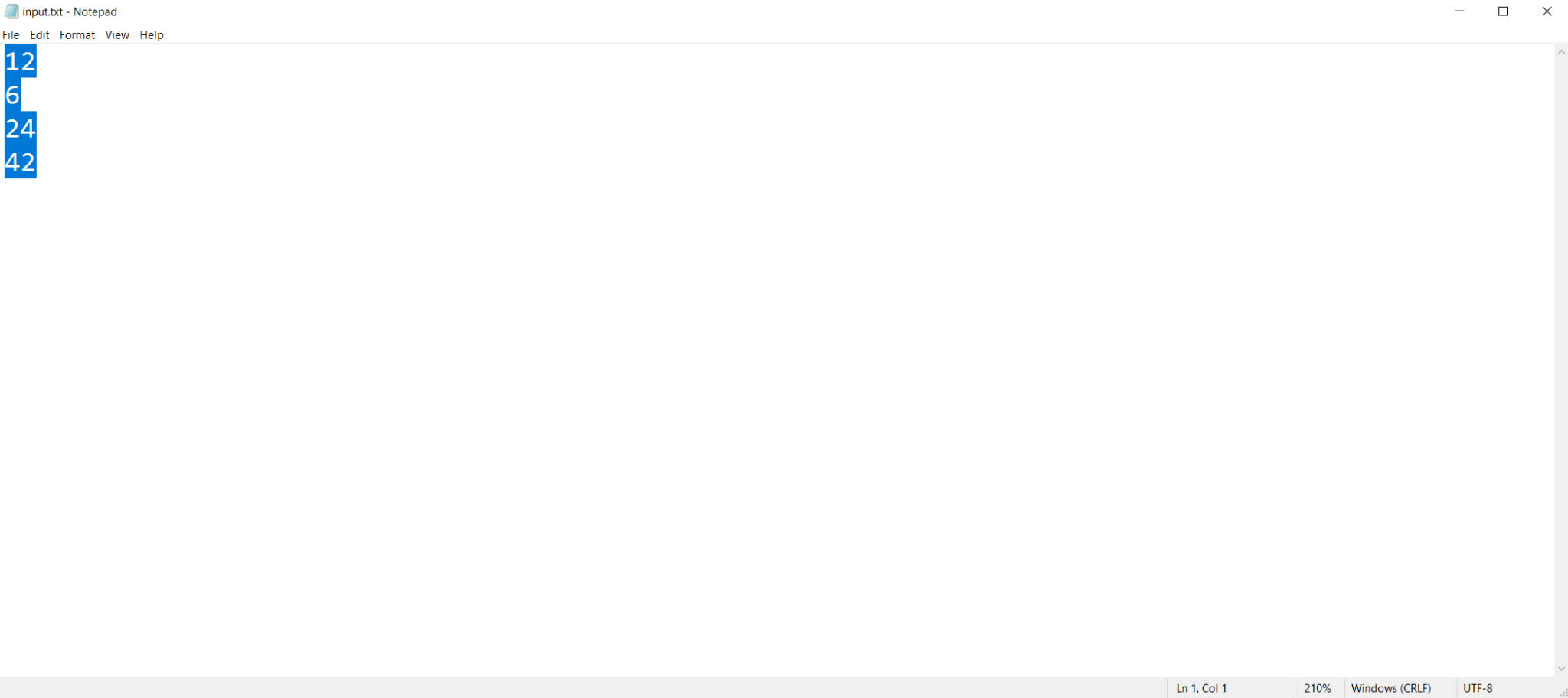
+ add(Fraction f) là phương thức trả về một đối tượng Fraction mới có được từ việc lấy tổng của fraction và f.

+ minus(Fraction f) là phương thức trả về một đối tượng Fraction mới có được từ việc lấy hiệu của fraction và f.

+ time(Fraction f) là phương thức trả về một đối tượng Fraction mới có được từ việc lấy tích của fraction và f.

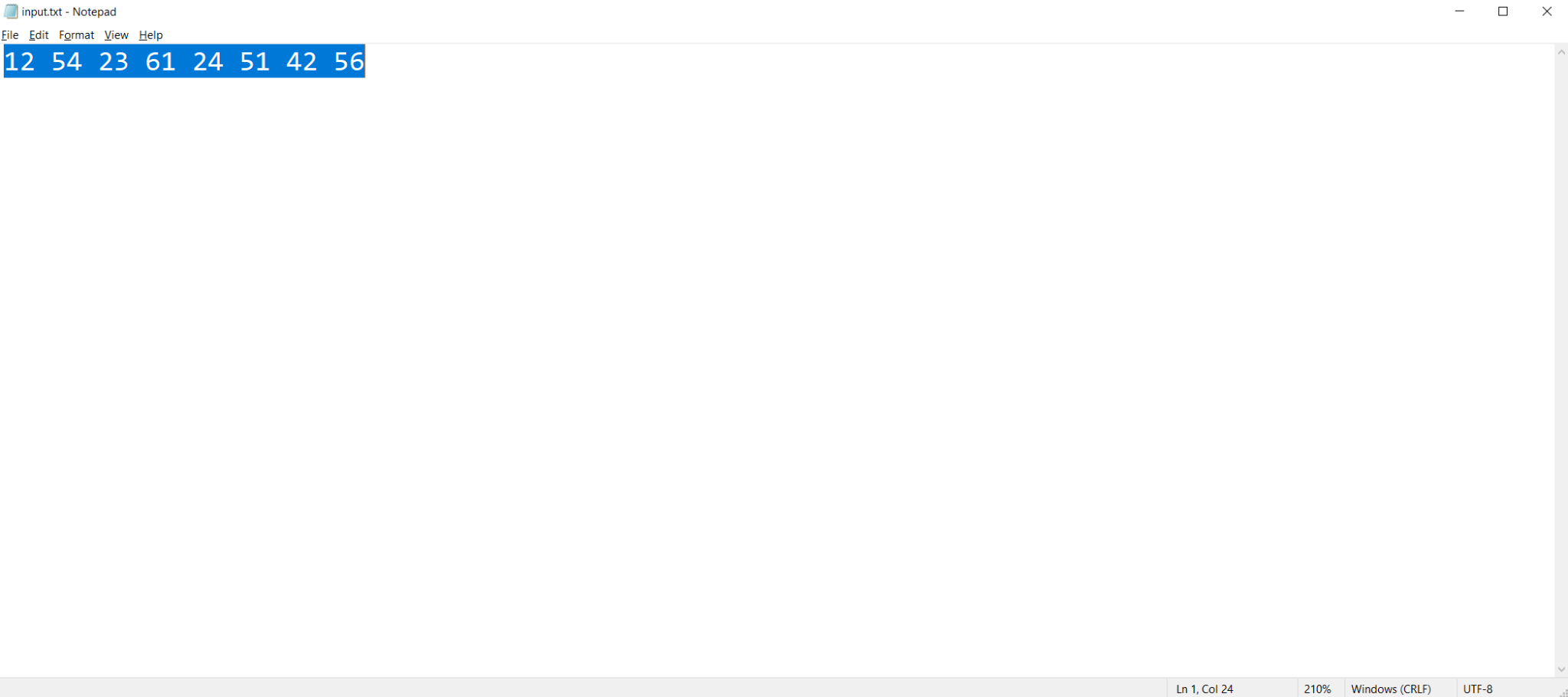
+ div(Fraction f) là phương thức trả về một đối tượng Fraction mới có được từ việc lấy thương của fraction và f.

19. Cho file input.txt có các dữ liệu sau:



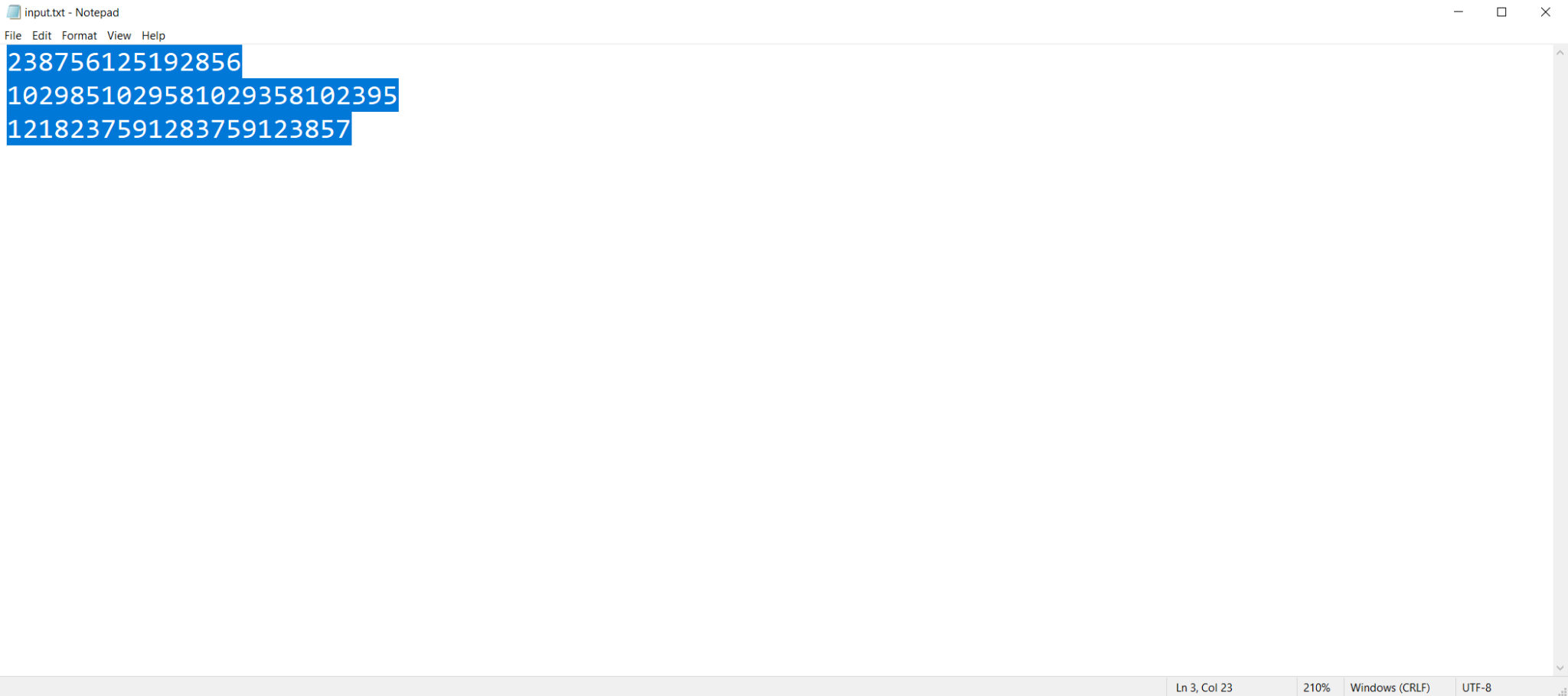
Biết trong file sẽ có n số nguyên > 0, mỗi số sẽ được ghi trên một dòng riêng biệt. Viết chương trình đọc dữ liệu từ file input.txt và tìm ra ước chung lớn nhất trong n số nguyên trong file input.txt.

20. Cho file input.txt có các dữ liệu sau:



Biết trong file sẽ có 1 dòng dữ liệu chứa n số nguyên > 0 cách nhau bằng dấu khoảng trắng. Viết chương trình đọc dữ liệu từ file input.txt và tìm ra số tất cả số nguyên tố trong n số nguyên trong file input.txt.

21. Cho file input.txt có các dữ liệu sau:



Biết trong file sẽ có 3 dòng dữ liệu tương ứng mỗi dòng là một chuỗi số. Viết chương trình đọc chuỗi số đầu tiên sau đó tính tổng các chữ số có trong chuỗi số đầu tiên và lưu kết quả lại, đọc chuỗi số thứ hai sau đó tính tích các chữ số lẻ có trong chuỗi số thứ hai và lưu kết quả lại, đọc chuỗi số thứ ba sau đó tính tổng các chữ số có trong chuỗi số thứ ba từ đó tính tích các chữ số lẻ có trong tổng vừa tìm được và lưu kết quả lại, cuối cùng ghi tất cả ba kết quả trên vào từng dòng trong file output.txt

22. Thực hiện các yêu cầu sau:

Tạo ra lớp Student:

- Trường:

+ int id: mã số sinh viên.

+ String name: tên sinh viên.

+ int age: tuổi sinh viên.

- Phương thức:

+ Student() là constructor mặc định.

+ Student(int id, String name, int age) là phương thức khởi tạo 3 trường id, name và age từ 3 tham số đầu vào.

+ getId(), setId(int id), getName(), setName(String name), getAge(), setAge(int age) là các getter và setter.

+ toString() là phương thức trả về chuỗi có dạng:

“Student [id=mã số sinh viên, name=tên sinh viên, age=tuổi sinh viên]”

Tạo ra lớp StudentManagement:

- Trường:

+ ArrayList<Student> studentList: danh sách sinh viên.

- Phương thức:

+ StudentManagement() là constructor mặc định, khởi tạo trường studentList = new ArrayList<Student>().

+ getStudentList(), setStudentList(ArrayList<Student> studentList) là các getter và setter.

+ create(Student s) là phương thức thêm đối tượng s vào danh sách sinh viên.

+ read() là phương thức in ra toàn bộ đối tượng Student đang quản lý có trong danh sách sinh viên, mỗi đối tượng sẽ được in ra trên một dòng.

+ update(Student s) là phương thức tìm ra trong danh sách sinh viên một đối tượng Student có id bằng với id của tham số đối tượng đầu vào s và cập nhật lại name và age của đối tượng đầu vào s cho đối tượng đó đã tìm được trong danh sách sinh viên.

+ delete(String id) là phương thức tìm ra đối tượng Student có id bằng với tham số đầu vào và xóa đi đối tượng đó ra khỏi danh sách sinh viên.