University of Science and Technology of Hanoi *** **Final Examination** **Subject: Algorithms and Data Structures** Sheet: 03 No of pages: 01		***Intake**: BI10* ***Academic year**: 2020–2021* ***Date**: 15/12/2020* ***Time**: 90 minutes* ***Important instructions*** *(according to lecturer's decision)* 1. Only the course slides and your own exercises' code are allowed in the examination venue. 2. Copy or using Internet will lead to heavy penalty	
Pathway coordinator		**Lecturer (or Head of Subject)**	**Dr. Đoàn Nhật Quang**
Student name		**Student's ID**	

Follow this instruction:
- Create a folder "ADS_YOURNAME_STUDENTID" in the Desktop.
*- Create the source files **question1.c** (or cpp) and **question2.c** for the corresponding problems.*
*- **Remove the executable files** (.exe) and **zip** all your source codes, send it to the address: **doan-nhat.quang@usth.edu.vn**.*
- Verify your name in the files and mails, un-named or incorrect-name files lead to 0.

Problem:
Given an array of 10 elements: 2, 10, 27, 13, 90, 45, 5, 26, 49, 50.
In this problem, we try to find all co-prime numbers. Two integer numbers are co-prime if the greatest common divisor of these two is 1.
The list of co-prime numbers from the above array is: {(2, 27), (2,13), (2,45), (2,5), (2,49), (10, 27), (10, 13), (10,49), (27,13), (27,5), (27,26), (27,50), (13,90), (13,45), (13,5), (13,49), (13,50), (90,49), (90,26), (90,49), (45,26), (45,49), (5,26), (5,49), (26,49), (49,50)}

Question 1: (10 pts)
- Write a program to find all co-prime numbers with a given array of integer numbers using **Iteration**. (8 pts)
- Calculate the complexity of your program (Best scenario, Worst scenario, Average). Justify your answer. (2 pts)

Question 2: (6 pts)
In this context, we try to find co-prime numbers using **at least a recursive function.**
- Implement the **recursive function** (2 pts).
- Write a program to solve the above Problem using **the recursive function** (4 pts)

Question 3: (4 pts)
- Write a program to solve the Problem using **Linked List ADT with necessary functions and Linked List principle**. (4 pts)
Note: Each element in the array is presented as a node in the **Linked List** data structure.

--END --