**Pattern Searching - Algorithms Assignments**

**Problem 1**

Learn and implement **Naive pattern-matching algorithm**

**Problem 2**

Learn and implement **Knuth-Morris-Pratt (KMP) pattern-matching algorithm**

**Problem 3**

Learn and implement **Boyer-Moore pattern-matching algorithm**

**Problem 4**

Địa chỉ kiểm soát truy cập phương tiện (địa chỉ MAC) là một định danh duy nhất được gán cho các giao diện mạng để truyền thông trên phân đoạn mạng vật lý.

Định dạng chuẩn (IEEE 802) để in địa chỉ MAC-48 ở dạng thân thiện với con người là sáu nhóm gồm hai chữ số thập lục phân (0 đến 9 hoặc A đến F), được phân tách bằng dấu gạch ngang (ví dụ: 01-23-45-67-89- AB).

Nhiệm vụ của bạn là kiểm tra chuỗi **inputString** đã cho xem nó **có tương ứng với địa chỉ MAC-48 hay không.**

**Ví dụ**

* Với inputString = "00-1B-63-84-45-E6", kết quả đầu ra sẽ là  
  isMAC48Address(inputString) = true;
* Với inputString = "Z1-1B-63-84-45-E6", kết quả đầu ra sẽ là  
  isMAC48Address(inputString) = false;
* Với inputString = "not a MAC-48 address", kết quả đầu ra sẽ là  
  isMAC48Address(inputString) = false.

**Input**:

* string inputString (15 ≤ inputString.length ≤ 20)

**Output**:

* + true nếu inputString tương ứng với quy tắc đặt tên địa chỉ MAC-48, false nếu không.

**Problem 5**

Một chuỗi hoàn hảo là một chuỗi gồm:

* Chỉ chứa các kí tự: ["x","y","z"]
* 2 ký tự liền kề không được giống nhau

Ví dụ: "xy", "xyx", "xyzx", .. là chuỗi hoàn hảo. "xxy", "yzz", "xzxx",.. là các chuỗi không hoàn hảo

Cho 2 số nguyên n và k hãy viết hàm **perfect\_string** để trả về một chuỗi thỏa mản yêu cầu sau:

* Chuỗi gồm n ký tự
* Trả về chuỗi thứ k trong danh sách được sắp xếp theo thứ tự từ điển. Trả về chuỗi trống nếu không tồn tại vị trí k có độ dài n

**Ví dụ:**

* Với n = 1 và k = 2 thì perfect\_string(n,k) = "y"

Giải thích: Ta sẽ có danh sách ký tự có độ dài n là: ["x", "y", "z"] và vị trí thứ 2 sẽ là "y"

* Với n = 1 và k = 5 thì perfect\_string(n,k) = ""

Giải thích: Ta sẽ có danh sách ký tự có độ dài n là: ["x", "y", "z"] và không tồn tại vị trí thứ 5 nên đầu ra sẽ bằng rỗng

**Input**: Integer **k, n**.

1 ≤ n ≤ 20, 1 ≤ k ≤ 1000

**Output**:String - Chuỗi thỏa mản yêu cầu đề bài

--------------------------- END --------------------------