|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BÀI** | **FILE CHƯƠNG TRÌNH** | **FILE DỮ LIỆU** | **FILE KẾT QUẢ** | **ĐIỂM** |
| **1** | **TILING.\*** | **TILING.INP** | **TILING.OUT** | **4** |
| **2** | **FAVORITENUM.\*** | **FAVORITENUM.INP** | **FAVORITENUM.OUT** | **5** |
| **3** | **BUTTONS.\*** | **BUTTONS.INP** | **BUTTONS.OUT** | **5** |
| **4** | **NEARNUMBERS.\*** | **NEARNUMBERS.INP** | **NEARNUMBERS.OUT** | **6** |

**Bài 1: Lát gạch sân nhà**

Một ngôi nhà có một sân rất rộng hình vuông với kích thước **N×N** . Sân này được chia thành **N** hàng và **N** cột, tạo thành các ô vuông nhỏ có kích thước bằng nhau. Chủ nhân của ngôi nhà là một người có niềm đam mê lớn với nghệ thuật. Vì vậy, ông nảy ra ý tưởng sẽ lát sân bằng những loại gạch độc đáo, mỗi loại gạch mang một nét đẹp riêng biệt.

Quy tắc lát gạch được định nghĩa như sau:

* Ô vuông ở vị trí hàng đầu tiên, cột đầu tiên sẽ được lát bằng viên gạch loại 1.
* Những ô vuông kề cạnh (phải, dưới) của viên gạch loại 1 sẽ được lát bằng viên gạch loại 2.
* Tiếp tục, những ô vuông kề cạnh với viên gạch loại 2 sẽ được lát bằng viên gạch loại 3, và quy trình này cứ thế tiếp diễn.

Nhiệm vụ của bạn là lập trình để mô phỏng quá trình lát gạch này và in ra trạng thái của mảnh sân sau khi đã hoàn thành lát gạch.

**Yêu cầu:**

* Nhập vào số **N** là kích thước sân. (1 ≤ N ≤ )
* In ra mảng **N x N** biểu diễn sân đã lát, trong đó mỗi ô chứa số tương ứng với loại gạch được sử dụng.

**Ví dụ:**

|  |  |
| --- | --- |
| **TILING.INP** | **TILING.OUT** |
| 5 | 1 2 3 4 5  2 3 4 5 6  3 4 5 6 7  4 5 6 7 8  5 6 7 8 9 |

**Ràng buộc:**

* 40% số điểm với 1 ≤ N ≤ 20
* 60% số điểm với 1 ≤ N ≤

**Bài 2: Chữ số yêu thích**

Trong toán học, **bội số** của một số là những số có thể chia hết cho số đó. Ví dụ, các bội số của 3 là: 3, 6, 9, 12,...

Hải và Linh là hai người bạn, mỗi người lại có sở thích đặc biệt với các bội số:

* Hải rất yêu thích các **bội số của số X**
* Linh lại đặc biệt yêu thích các **bội số của số Y**

Giờ đây, cả hai muốn biết có bao nhiêu số mà họ yêu thích, tức là các số là bội số của **X** hoặc **Y**, trong phạm vi từ 1 đến **N**.

Nhiệm vụ của bạn là viết một chương trình tính toán số lượng các số là số yêu thích của Hải hoặc của Linh trong khoảng từ 1 đến **N**

**Yêu cầu:**

* Nhập vào ba số nguyên **N**, **X** và **Y** (1 ≤ **X**, **Y** ≤ , 1 ≤ **N** ≤ )
* Tính và in ra số lượng các số yêu thích của Hải hoặc của Linh trong khoảng từ 1 đến **N**

**Ví dụ:**

|  |  |
| --- | --- |
| **FAVORITENUM.INP** | **FAVORITENUM.OUT** |
| 3 7 15 | 7 |

Kết quả: **7 số**. Các số yêu thích của Hải hoặc Linh là 3, 6, 7, 9, 12, 14, 15

**Ràng buộc:**

* 20% số điểm với 1 ≤ X, Y ≤ 10, 1 ≤ N ≤ 100
* 30% số điểm với 1 ≤ X, Y ≤ , 1 ≤ N ≤
* 50% số điểm với 1 ≤ X, Y ≤ , 1 ≤ N ≤

**Bài 3: Những chiếc khuya áo**

Một thợ may có nhiều chiếc khuy áo, mỗi chiếc khuy được đánh dấu bằng một ký tự trong khoảng từ **'a' đến 'z'**. Thợ may nhận thấy rằng có rất nhiều cách để sắp xếp các khuy áo thành một hàng ngang.

Khi nghiên cứu các cách sắp xếp, ông phát hiện ra rằng một số cách xếp sẽ tạo thành các đoạn con lặp lại liên tiếp trong chuỗi, những đoạn này sẽ lặp lại cho tới hết chuỗi. Ví dụ, nếu ông có 4 chiếc khuy áo lần lượt là "abba", và sắp xếp chúng theo thứ tự "abab", ông sẽ nhận được một chuỗi có đoạn con lặp lại là "ab". Trong trường hợp này, đoạn "ab" lặp lại 2 lần liên tiếp.

Ông thợ may muốn tìm cách sắp xếp để chuỗi tạo thành có **đoạn con lặp lại liên tiếp nhiều lần nhất**. Nếu có nhiều cách sắp xếp thỏa mãn số lần lặp lại lớn nhất, ông muốn chọn cách sắp xếp mà đoạn con lặp lại có thứ tự từ điển nhỏ nhất.

**Yêu cầu:**

- Nhập vào một chuỗi ký tự **S** biểu thị các khuy áo, (|s| ≤ )

- In ra chuỗi khuya áo được lặp lại trong cách sắp xếp sao cho có nhiều đoạn con lặp lại nhất, lưu ý rằng nếu có nhiều cách sắp xếp có cùng số lượng đoạn con lặp lại nhều nhất, sẽ chọn cách sắp xếp để đoạn con đó có thứ tự từ điển bé nhất

**Ví dụ:**

|  |  |
| --- | --- |
| **BUTTONS.INP** | **BUTTONS.OUT** |
| baabbaab | ab |

Giải thích: Chuỗi khuya áo có thể sắp xếp thành **abababab** lặp lại **ab** 4 lần

**Ràng buộc:**

* 30% số điểm với độ dài chuỗi **S**: 1 ≤ |**S**| ≤ 10
* 70% số điểm với độ dài chuỗi **S**:, 1 ≤ |**S**| ≤

**Bài 4: Số xấp xỉ**

Một số nguyên được gọi là số xấp xỉ nếu tất cả các chữ số liên tiếp trong số đó có giá trị chênh lệch đúng 1 đơn vị.

**Ví dụ:**

* Các số 1234, 12123, 1210 là số xấp xỉ.
* Các số 112, 12345, 2109 không phải là số xấp xỉ.

**Yêu cầu:**

* Nhập vào số nguyên dương N ((1 ≤ N ≤ )
* Hãy tìm tất cả các số xấp xỉ nhỏ hơn hoặc bằng **N** theo thứ tự tăng dần

Ví dụ:

|  |  |
| --- | --- |
| **NEARNUMBERS.INP** | **NEARNUMBERS.OUT** |
| 22 | 1 2 3 4 5 6 7 8 9 10 12 21 |

**Ràng buộc:**

* 30% số điểm với: 1 ≤ N ≤
* 70% số điểm với: 1 ≤ N ≤

**Hướng dẫn chấm**

**Bài 1: Lát gạch sân nha**

**Subtask 1: 40% số điểm (3 for lồng nhau)**

Sử dụng 3 vòng for lồng nhau, làm theo yêu cầu của đề, tìm lần lượt các số từ 1, 2, 3, 4, … và điền vào mảng

Nghĩa là sẽ điểm a[i][j] = k, nếu a[i – 1][j] hoặc a[i][j – 1] bằng k – 1.

**Subtask 2: 60% số điểm (2 for lồng nhau)**

Nhận định rằng loại gạch ở hàng i và cột j sẽ có giá trị là i+ j – 1, nên chị cần 2 vòng for lồng nhau (i, j) cùng chạy từ 1 đến n, mỗi ô in ra luôn giá trị của i + j - 1

**Bài 2: Chữ số yêu thích**

**Subtask 1: 20% số điểm (Kiểm tra bội từng số)**

Dùng biến dem để lưu số lượng bội của **x** hoặc **y**. Duyệt từng bội số của **x** và **y** như yêu cầu của đề, nếu bội số nào nằm trong khoảng 1 đến **n** thì tăng **dem** thêm 1.

Cách này vẫn có khả năng bị trùng những số vừa là bội của **x** vừa là bội của **y**.

**Subtask 1: 30% số điểm (Kiểm tra từng số)**

Dùng biến dem để lưu số lượng bội của **x** hoặc **y**. Dùng vòng lặp chạy từ 1 đến **n**, kiểm tra xem thử các số trong vòng lặp có chia hết cho **x** hoặc **y** không, nếu chia hết thì tăng **dem** thêm 1

Cách làm này sẽ đúng được 50% số điểm (đúng luôn subtask1)

**Subtask 1: 50% số điểm (Sử dụng công thức)**

Tù 1 đến n sẽ có **[n / x]** số là bội của **x,** **[n / y]** số là bội của **y**. Tuy nhiền cần trừ đi những số vừa là bội của **x**, vừa là bội của **y** là **[n / lcm(x, y)]**

Công thức là **[n / x] + [n / y] - [n / lcm(x, y)]**

**Bài 3: Những chiếc khuya áo**

**Subtask 1: 30% số điểm (Vét cạn)**

Kiểm tra tất cả những hoán vị của chuỗi **S**, mỗi chuỗi đưa ra đoạn con lặp lại ngắn nhất (Vì đề muốn lặp lại nhiều lần nhất nên đoạn con được lặp phải ngắn nhất)

Tìm ra chuỗi ngắn nhất và thứ tự từ điển nhỏ nhất để in ra

**Subtask 2: 30% số điểm (Mảng đánh dấu, tìm ước chung)**

Dùng mảng đánh dấu đếm lần lượt số lượng của ký tự từ **‘a’** đến **‘z’** trong chuỗi.

Tìm ra số nhóm tối đa có thể chia được từ danh sách số lượng các ký tự, bằng cách tìm ước chung lớn nhất của danh sách số lần xuất hiện các ký tự. Ví dụ đặt nó là **K** nhóm

In ra các ký tự từ **‘a’** đến **‘z’** với số lần là **m[ch] / K**

(**m[ch]** là số lần xuất hiện của ký tự **ch** trong chuỗi **S**)

**Bài 4: Số xấp xỉ**

**Subtask 1: 30% số điểm (Vét cạn)**

Dùng biến **dem** để đếm số xấp xỉ từ 1 đến **n**

Cách kiểm tra số **x** của phải số xấp xỉ không: dùng mod và div của 10 để tách các chuỗi số của **x** ra, kiểm tra sự chệnh lệch giữa 2 chữ số liền kề, nếu tất cả các cặp liền kề đều lệch nhau 1 đơn vị thì **x** là số xấp xỉ.

Áp dụng lên tất cả các số từ 1 đến **n**

**Subtask 1: 30% số điểm (Phương pháp sinh)**

Sử dụng phương pháp sinh để sinh ra các số xấp xỉ, từ số xấp xỉ trước đó

Ví dụ **X** là một số xấp xỉ. Ta sinh ra cách số xấp xỉ bằng cách sau

* Tìm **K** là chữ số cuối cùng của **X**
* Sinh ra số xấp xỉ mới là **X \* 10 + (K – 1)**, nếu K ≥ 0, và số xấp xỉ **X \* 10 + (K + 1**) nếu K ≤ 9

Ví dụ số 123, sẽ sinh được ra 2 số xấp xỉ mới là 1231 và 1234

Tạo danh sách số xấp xỉ ban đầu từ 1 đến 9, áp dụng phương pháp sinh trên cho các số cho đến khi vượt quá **N**