***Cho số tự nhiên product  
Hãy tìm số nguyên dương nhỏ nhất (lớn hơn 0) mà tích các chữ số của số đó bằng product. Nếu không có số thỏa mãn, trả ra -1.***

**Ví dụ**

* Với product = 12, thì kết quả digitsProduct(product) = 26;
* Với product = 19, thì kết quả digitsProduct(product) = -1.

**Đầu vào/đầu ra:**

* **[Thời gian chạy] 0.5 giây**
* **[Đầu vào] integer product** *Điều kiện:*  
  0 ≤ product ≤ 600.
* **[Đầu ra] integer**

**Gợi ý:**

* Chọn các chữ số của số cần tìm là ước số của product
* Các chữ số lớn ở phía sau, các chữ số nhỏ ở phía trước (Ví dụ 129 thay vì 921)

**Lý thuyết :**

* Số A nhỏ hơn số B khi :
  + số chữ số của A ít hơn số chữ số của B
  + số chữ số của A bằng của B, và vị trí đầu tiên (từ trái sang) khác nhau của 2 số, chữ số của A có giá trị bé hơn
* Vậy, để thu được số nhỏ nhất, ta cần tối ưu về số lượng chữ số, rồi sau đó là thứ tự các chữ số
  + Ta lấy theo cách sau : lấy chữ số to nhất lớn hơn 1 mà product chia hết, đặt ở cuối, rồi tiếp tục làm thế với product mới (sau khi đã chia cho chữ số đó)
    - **TH1 :**product = 1 ⇒ Ta đã tách được product thành mọi chữ số, nên sẽ có kết quả cho bài toán
    - **TH2 :**product > 1. Lúc này product sẽ gồm các số nguyên tố > 9, nên ko còn cách nào có thể tách ra nữa. Ko có số nào thỏa mãn đề bài trong trường hợp này
  + Tính đúng đắn :
    - Giá trị ta tách ra càng to, thì càng tách ra ít lần. Nên số chữ số chắc chắn là tối ưu nhất
    - Giả sử có nhiều cách tách ra có cùng số chữ số, thì việc tách ra chữ số càng to, sẽ càng tốt hơn cho phần ở trước
  + **Ví dụ** **:** product = 108 ⇒ 9 \* 6 \* 2 ⇒ kết quả : 269
    - Ta nên lấy { 9, 6, 2 } thay vì { 2, 2, 3, 3, 3 } để tối ưu về số chữ số
    - Ta nên lấy { 9, 6, 2 } thay vì { 9, 4, 3 } vì sau khi lấy 6, product còn lại là 2 thay vì 3. Và 2 tốt hơn khi đặt ở *hàng đầu tiên* (từ trái sang)
* **Bài toán tương tự** : Cho số product. Tìm số có tổng các chữ số nhỏ nhất. Nếu có nhiều số thỏa mãn, in ra số bé nhất.

**Hướng dẫn bài tập.**

Để kết quả là bé nhất ta phải đảm bảo rằng:

* kết quả ít chữ số nhất có thể, ví dụ 3233 ta có thể thay thế bằng 69.
* Các số của kết quả sắp xếp không giảm.

Nên ta số lần lượt chia product cho các số từ 9 đến 2.

Lưu ý: bài này có một số trường hợp đặc biệt.

Code mẫu:

Ngôn ngữ C++:

int digitsProduct(int product)

{

if (product == 1) return 1;

if (product == 0) return 10;

int ans = 0;

for (int i = 9; i >= 2; i--){

while(product % i == 0){

ans = ans \* 10 + i;

product /= i;

}

}

// số ans bây giờ là kết quả, nhưng nó đang bị ngược.

int ans2 = 0;

while (ans > 0){

ans2 = ans2 \* 10 + ans % 10;

ans /= 10;

}

return (product == 1) ? ans2 : -1;

}