Pangrams

Roy wanted to increase his typing speed for programming contests. So, his friend advised him to type the sentence "The quick brown fox jumps over the lazy dog" repeatedly, because it is a *pangram*. (Pangrams are sentences constructed by using every letter of the alphabet at least once.)

After typing the sentence several times, Roy became bored with it. So he started to look for other pangrams.

Given a sentence , tell Roy if it is a pangram or not.

Input Format

Input consists of a string .

Constraints

Length of can be at most   and it may contain spaces, lower case and upper case letters. Lower-case and upper-case instances of a letter are considered the same.

Output Format

Output a line containing *pangram* if is a pangram, otherwise output *not pangram*.

Sample Input

Input #1

```
We promptly judged antique ivory buckles for the next prize
```

Input #2

```
We promptly judged antique ivory buckles for the prize
```

Sample Output

Output #1

```
pangram
```

Output #2

```
not pangram
```

Explanation

In the first test case, the answer is *pangram* because the sentence contains all the letters of the English alphabet.

Solution

```java
public static void main(String args[] ) throws Exception {

    String string;
    Scanner sc = new Scanner(System.in);
    string = sc.nextLine();

    string = string.toUpperCase();
    char array[] = new char[26];
    int array2[]= new int[26];

    for(int i=0; i<string.length(); i++)
       {
            char temp = string.charAt(i);
            int convert = (int)temp;
            if(convert<=90 && convert>=65)
                {
                     convert-=65;
                     array2[convert]+=1;
                }
       }

    //traversing the array
    int flag=0;
    for(int i=0; i<array2.length; i++)
       {
            if(array2[i]<=0)
               {
                   flag=1;
                   break;
               }
       }


if(flag==0){System.out.println("pangram");}else{System.out.println("not pangram");}
```

`'Anshul AgGarwal