1. How many different ways you can find the no. of characters in the string,

answers : using for loop and duplicate flag, or using Hashmap (both the approach is good, but Hashmap is more accurate.)

2. Write a program to reverse a number in any language.

Answer: One of the possible answer

public class reverseNumber {

public long reverse(long num)

{

long temp=0;

while(num!=0)

{

temp=(temp\*10)+(num%10);

num=num/10;

}

return temp;

}

public static void main(String args[])

{

long n= 342619;

reverseNumber inp = new reverseNumber();

System.out.println(“Given number is “+ n);

System.out.println(“Reverse of given number is “+inp.reverse(n));

}

}

3.PanGram based questions:

Note : A pangram is a sentence containing every letter in the English Alphabet (a-z).

Pangram example: The quick brown fox jumps over the lazy dog

approach :

Create an boolean array that can hold 26 boolean corresponding to each alphabet in English.

Traverse each character of the sentence and mark corresponding index (a=0, b=1,.. z=26) in the boolean array. Take care of upper and lower case.

Traverse through the entire boolean array and see if all characters are covered. If all positions are set to true then input sentence is a pangram.

public class Pangram {

public boolean checkPangram(String str) {

boolean[] mark = new boolean[26];

int index = 0;

for (int i = 0; i < str.length(); i++) {

// If uppercase character, subtract 'A' to find index.

if ('A' <= str.charAt(i) && str.charAt(i) <= 'Z')

index = str.charAt(i) - 'A';

else if ('a' <= str.charAt(i) && str.charAt(i) <= 'z')

index = str.charAt(i) - 'a';

mark[index] = true;

}

for (int i = 0; i <= 25; i++)

if (!mark[i])

return (false);

return (true);

}

}

There is also an optimized version of this problem:

same program can use an integer (has 32 bits) to hold the state of 26 alphabets by use of Bitwise OR and left shift operator.

private void pangramEfficient() {

String s = "The quick brown fox jumps over the lazy dog";

int i = 0;

for (char c : s.toCharArray()) {

int x = Character.toUpperCase(c);

if (x >= 'A' && x <= 'Z') {

i |= 1 << (x - 'A');

}

}

if (i == (1 << (1 + 'Z' - 'A')) - 1) {

System.out.println("input is a pangram");

} else {

System.out.println("input is not a pangram");

}

}

4. Questions on Combinations based on day to day activity:

Today is Sunday and Disha is out for the shopping. She has reached to the famous market of the city. She has a list of items which she wants to buy but she has a very low amount of cash today and no shop accepts any online payment method in the market. Disha has only D rupees with her and wants to buy N items (she wants to buy as many units of item as possible). She wants to shop in a way such that first she buy 1 quantity of all the items she wants to buy and then she will move to buy 2nd quantity of any item.

Can you help her by calculating how many items she can buy today if she shops optimally?

You are given an array which represents the price of items Disha wants to buy.

Example :-

If she wants to buy 4 items having price 4, 6, 2, 5 respectively and has 49 rupees only then she will buy in this order:- 1st item, 2nd item, 3rd item, 4th item (It will cost 17 rupees). Again she will buy in this order 1st, 2nd, 3rd, 4th (it will cost 17 rupees). Now, she has remaining 15 rupees so she can buy any 3 items.

Input Format

The first line of input consists of two space-separated integers, N (number of items Disha wants to buy) and D (amount of cash Disha has with her).

The second line of input consists of N space-separated integers which represents the price of the N items Disha wants to buy.

Constraints

1<= N <=1e9

1<= Ni <=10^15 (1e15)

1<= D <=10^18 (1e18)

Output Format

Print the count of items which Disha can buy.

Sample TestCase 1

Input

5 24

4 8 9 7 2

Output

4