American Computer Science League

2022-2023 ● Contest 2: Binary Counting ● Intermediate Division

PROBLEM: Given a string of characters found on the keyboard, convert each character in the string to the binary equivalent of its ASCII code. In the resulting concatenated string, search for the increasing sequence of binary numbers starting with 0, 1, 10, 11, … until a number cannot be found anywhere in the string. Look from the start of the string. If the binary number is found, remove that occurrence of the binary number from the string. Then look from the end of the string. If the binary number is found, remove that occurrence of the binary number from the string. When the binary number cannot be found at all, output the decimal equivalent of the last binary number that can be found.

EXAMPLE: For the string "Roses are red.", convert it to a concatenated string of binary numbers using each character's ASCII code as follows:

Char	ASCII	Binary
R	82	01010010
o	111	01101111
s	115	01110011
e	101	01100101
s	115	01110011
sp	32	00100000
a	97	01100001

Char	ASCII	Binary
r	114	01110010
e	101	01100101
sp	32	00100000
r	114	01110010
e	101	01100101
d	100	01100100
.	46	00101110

Now search for binary numbers beginning with 0 in the following string:

01010010 01101111 01110011 01100101 01110011 00100000 01100001
01110010 01100101 00100000 01110010 01100101 01100100 00101110

Remove the 0 from both ends so the string becomes:

1010010 01101111 01110011 01100101 01110011 00100000 01100001
01110010 01100101 00100000 01110010 01100101 01100100 00101111

Remove the 1 from both ends so the string becomes:

010010 01101111 01110011 01100101 01110011 00100000 01100001
01110010 01100101 00100000 01110010 01100101 01100100 001011

Remove 10 from both ends so the string becomes:

0010 01101111 01110011 01100101 01110011 00100000 01100001
01110010 01100101 00100000 01110010 01100101 01100100 0011

Remove 11 from both ends so the string becomes:

0010 001111 01110011 01100101 01110011 00100000 01100001
01110010 01100101 00100000 01110010 01100101 01100100 00

Continuing until we search for 1001, the resulting string is:

0000110101011100110000011000010111001001100100000000001000100

The string 1010 can only be found once from the start of the string so the resulting string is:

0000110111001100000110000101110010011001000000000001000100

The process continues until the final string becomes:

0000110000011000010010010000000001000100

The binary number 1101 cannot be found in the final string. The binary numbers 0, 1, 10, 11, 100, 101, 110, 111, 1000, 1001, 1010, 1011, and 1100 were found and deleted from one or both sides of the string, but not 1101. The binary number 1100 is the last one found in the string so the output is its decimal equivalent which is 12.

INPUT: A string, *s*, containing any characters found on the keyboard. The string will be fewer than 200 characters.

OUTPUT: For each input, output an integer representing the decimal equivalent of the last binary number that can be found in the string after all deletions have been made.

SAMPLE INPUT:

```
1. Roses are red.
2. A is Alpha; B is Bravo; C is Charlie.
3. A stitch in time saves nine.
4. 1, 2: Buckle my shoe! 3, 4: Shut the door!
5. Is HackerRank the platform used by ACSL?
```

SAMPLE OUTPUT:

```
1. 12
2. 20
3. 14
4. 22
5. 27
```