Problem Set 2 Exercise #25: Prime Products

Reference: Lecture 5 notes

Learning objective: Algorithm design

Estimated completion time: 60 minutes

Problem statement:

[Past year CS1101 Practical Exam Question]

A non-zero integer can be expressed as a product of some prime numbers arranged in ascending order. We will include 1 in the expression. For example, 20 can be expressed as $1 \times 2 \times 2 \times 5$; 882 can be expressed as $1 \times 2 \times 3 \times 3 \times 7 \times 7$; and 131 can be expressed as 1×131.

For negative integers, we include -1 in the expression. For example, -731 is $-1 \times 17 \times 43$; and -36 is $-1 \times 2 \times 2 \times 3 \times 3$.

Write a program **prime_product.c** that accepts a non-zero integer and displays the expression as illustrated above.

Some tips are provided on the next page.

Sample run #1:

```
Enter integer: 1
1 = 1
```

Sample run #2:

```
Enter integer: 992
992 = 1 * 2 * 2 * 2 * 2 * 2 * 31
```

Sample run #3:

```
Enter integer: -731
-731 = -1 * 17 * 43
```

Useful tips:

To get started, you may ignore the negative value case and focus on the positive value first. After that, you can then work on the negative value case, which you will discover is just a simple extension, if you have done things right.

You might think that this exercise involves finding the next prime number as you roll out the "factorization" – actually there is no such need.