

WORKED EXAMPLE 2.2

Computing the Cost of Stamps



You are asked to simulate a postage stamp vending machine. A customer inserts dollar bills into the vending machine and then pushes a “purchase” button. The vending machine gives out as many first-class stamps as the customer paid for, and returns the change in penny (one-cent) stamps. A first-class stamp cost 44 cents at the time this book was written.

Step 1 Understand the problem: What are the inputs? What are the desired outputs?

In this problem, there is one input:

- The amount of money the customer inserts

There are two desired outputs:

- The number of first-class stamps the machine returns
- The number of penny stamps the machine returns

Step 2 Work out examples by hand.

Let’s assume that a first-class stamp costs 44 cents and the customer inserts \$1.00. That’s enough for two stamps (88 cents) but not enough for three stamps (\$1.32). Therefore, the machine returns two first-class stamps and 12 penny stamps.

Step 3 Write pseudocode for computing the answers.

Given an amount of money and the price of a first-class stamp, how can you compute how many first-class stamps can be purchased with the money? Clearly, the answer is related to the quotient

$$\frac{\text{amount of money}}{\text{price of first-class stamp}}$$

For example, suppose the customer paid \$1.00. Use a pocket calculator to compute the quotient: $\$1.00/\$0.44 \approx 2.27$.

How do you get “2 stamps” out of 2.27? It’s the quotient without the remainder. In C++, this is easy to compute if both arguments are integers. Therefore, let’s switch our computation to pennies. Then we have

$$\text{number of first-class stamps} = 100 / 44 \text{ (integer division, without remainder)}$$

What if the user inputs two dollar? Then the numerator becomes 200. What if the price of a stamp goes up? A more general equation is

$$\text{number of first-class stamps} = 100 \times \text{dollars} / \text{price of first-class stamps in cents}$$

How about the penny stamps that are returned as change? Look at it this way. The change is the customer payment, reduced by the value of the first-class stamps purchased. In our example, the customer is due 12 cents worth of penny stamps the difference between 100 and 2×44 . Here is the general formula:

$$\text{penny_stamps} = 100 \times \text{dollars} - \text{number of first-class stamps} \times \text{price of first-class stamp}$$

Step 4 Declare the variables and constants that you need, and specify their types.

Here, we have three variables:

- dollars
- first_class_stamps
- penny_stamps

There is one constant, FIRST_CLASS_STAMP_PRICE.

The variable `dollars` and constant `FIRST_CLASS_STAMP_PRICE` must be of type `int` because the computation of `first_class_stamps` uses integer division. The remaining variables are also integers, counting the number of first-class and penny stamps. Thus, we have

```
const int FIRST_CLASS_STAMP_PRICE = 44; // Price in pennies
int dollars; // Filled through input statement

int first_class_stamps = 100 * dollars / FIRST_CLASS_STAMP_PRICE;
int penny_stamps = 100 * dollars - first_class_stamps * FIRST_CLASS_STAMP_PRICE;
```

Step 5 Turn the pseudocode into C++ statements.

Our computation depends on the number of dollars that the user provides. Translating the math into C++ yields the following statements:

```
first_class_stamps = 100 * dollars / FIRST_CLASS_STAMP_PRICE;
penny_stamps = 100 * dollars - first_class_stamps * FIRST_CLASS_STAMP_PRICE;
```

Step 6 Provide input and output.

```
cout << "Enter number of dollars: ";
cin >> dollars;
```

When the computation is finished, we display the result.

```
cout << "First class stamps: " << setw(6) << first_class_stamps << endl
     << "Penny stamps:      " << setw(6) << penny_stamps << endl;
```

Step 7 Include the required headers and provide a main function.

We need the `<iostream>` header for all input and output. Because we use the `setw` manipulator, we also require the `<iomanip>` header.

Here is the complete program, `ch02/stamps.cpp`:

```
#include <iostream>
#include <iomanip>

using namespace std;

int main()
{
    const int FIRST_CLASS_STAMP_PRICE = 44; // Price in pennies

    cout << "Enter number of dollars: ";
    int dollars;
    cin >> dollars;

    int first_class_stamps = 100 * dollars / FIRST_CLASS_STAMP_PRICE;
    int penny_stamps = 100 * dollars - first_class_stamps * FIRST_CLASS_STAMP_PRICE;
    cout << "First class stamps: " << setw(6) << first_class_stamps << endl
         << "Penny stamps:      " << setw(6) << penny_stamps << endl;

    return 0;
}
```

Program Run

```
Enter number of dollars: 4
First class stamps:      9
Penny stamps:           22
```