p5.17

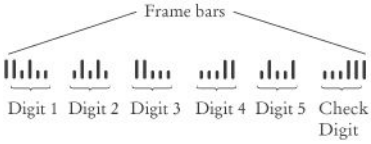
Henry Hsu

October 11, 2011

1 Specification

Postal bar codes. For faster sorting of letters, the United States Postal Service encourages companies that send large volumes of mail to use a bar code denoting the zip code (see Figure 10).

The encoding scheme for a five-digit zip code is shown in Figure 11. There are full-height frame bars on each side. The five encoded digits are followed by a check digit, which is computed as follows: Add up all digits, and choose the check digit to make the sum a multiple of 10. For example, the zip code 95014 has a sum of 19, so the check digit is 1 to make the sum equal to 20.



Each digit of the zip code, and the check digit, is encoded according to the following table where 0 denotes a half bar and 1 a full bar.

	7	4	2	1	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	0
6	0	1	1	0	0
7	1	0	0	0	1
8	1	0	0	1	0
9	1	0	1	0	0
0	1	1	0	0	0

The digit can be easily computed from the bar code using the column weights 7, 4, 2, 1, 0. For example, 01100 is

$$0 \times 7 + 1 \times 4 + 1 \times 2 + 0 \times 1 \times 0 \times 0 = 6$$

. The only exception is 0, which would yield 11 according to the weight formula.

Write a program that asks the user for a zip code and prints the bar code. Use : for half bars, | for full bars. For example, 95014 becomes

|| : | ::: | : | : || :::::: || : | :: | ::: |||

2 Analysis/Design

- Get zipcode
- Split up the zip code into 5 individual digits
- calculate the check digit
- display the bar code associated with the 5 individual zip digits

3 Implementation

"p5_17.cpp" 3≡

⟨ *Include files* 5b ⟩
⟨ *get zip code* 5a ⟩
⟨ *print bar code* 4a ⟩
⟨ *create check digit* 4b ⟩

```
int main()
{
        int zip_code = get_zip_code();

        cout << "For Zip code " << zip_code << ", the associated bar code is: " << endl;


        /* This code block splits up the zip code into individual integers */
        int zip1 = 0, zip2 = 0, zip3 = 0 , zip4 = 0, zip5 = 0, zip_sum;
        int zip_copy = zip_code;
        zip5 = zip_copy % 10;
        zip_copy = zip_copy /= 10;
        zip4 = zip_copy % 10;
        zip_copy = zip_copy /= 10;
        zip3 = zip_copy % 10;
        zip_copy = zip_copy /= 10;
        zip2 = zip_copy % 10;
        zip_copy = zip_copy /= 10;
        zip1 = zip_copy;
        zip_sum = (zip1 + zip2+ zip3 + zip4 + zip5);


        /* Calls the check digit creater function */
        int check_digit = create_check_digit(zip_sum);

        cout << "|";
        print_bar_code(zip1);
        print_bar_code(zip2);
        print_bar_code(zip3);
        print_bar_code(zip4);
        print_bar_code(zip5);
        print_bar_code(check_digit);
        cout << "|";
}
```

◇

⟨ *print bar code* 4a ⟩ ≡

```
void print_bar_code(int zip_digit)
{
        if (zip_digit == 1)
                cout << ":::||";
        if (zip_digit == 2)
                cout << "::|:|";
        if (zip_digit == 3)
                cout << "::||:";
        if (zip_digit == 4)
                cout << ":|::|";
        if (zip_digit == 5)
                cout << ":|:|:";
        if (zip_digit == 6)
                cout << ":||::";
        if (zip_digit == 7)
                cout << "|:::|";
        if (zip_digit == 8)
                cout << "|::|:";
        if (zip_digit == 9)
                cout << "|:|::";
        if (zip_digit == 0)
                cout << "||:::";
}
```

◇

Fragment referenced in 3.

⟨ *create check digit* 4b ⟩ ≡

```
int create_check_digit(int zip_sum)
{
        int check_digit = 0;
        check_digit = (10 - (zip_sum % 10));
        if ((zip_sum % 10) < 1)
                check_digit = 0;
        return check_digit;
}
```

◇

Fragment referenced in 3.

⟨ *get zip code* 5a ⟩ ≡

```
int get_zip_code()
{
        int zip_code = 0;
        cout << "Please enter a 5 digit zip code: ";
        cin >> zip_code;
        if ( cin.fail() )                               // Ensures integer input
        {
                cin.clear();
                cout << "Please rerun the program and only enter a 5 digit zip code. ";
                exit(1);
        }
        if(zip_code > 99999 || zip_code < 10000)        // Ensures 5 digit input
        {
                cin.clear();
                cout << "Please rerun the program and only enter a 5 digit zip code. ";
                exit(1);
        }
        return zip_code;
}
```

◇

Fragment referenced in 3.

These are the include files needed for library function calls

⟨ *Include files* 5b ⟩ ≡

```
        #include <iostream>
        #include <cstdlib>
        using namespace std;
```

◇

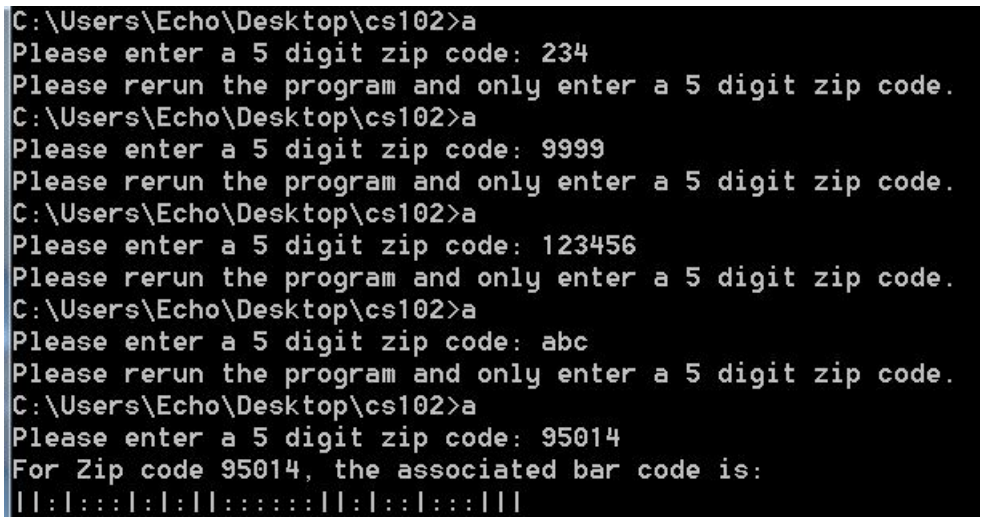
Fragment referenced in 3.

4 Test

A test value of 95014 should yield

||:|:::|:|:||::::::||:|::|:::|||

as the bar code


C:\Users\Echo\Desktop\cs102>a
Please enter a 5 digit zip code: 234
Please rerun the program and only enter a 5 digit zip code.
C:\Users\Echo\Desktop\cs102>a
Please enter a 5 digit zip code: 9999
Please rerun the program and only enter a 5 digit zip code.
C:\Users\Echo\Desktop\cs102>a
Please enter a 5 digit zip code: 123456
Please rerun the program and only enter a 5 digit zip code.
C:\Users\Echo\Desktop\cs102>a
Please enter a 5 digit zip code: abc
Please rerun the program and only enter a 5 digit zip code.
C:\Users\Echo\Desktop\cs102>a
Please enter a 5 digit zip code: 95014
For Zip code 95014, the associated bar code is:
||:|:::|:|:||::::::||:|::|:::|||