

System Manual:-

```
/
*****
*****
PROGRAM : postal_bar_code_converter.cpp
PURPOSE : This program will convert any five digit zipcode into its complete barcode. For this
reason,
    it will take the integer value of the zipcode as an input, and will show the barcode as an output.
    Here, the program will ask its user to input a zipcode, the program will convert it into barcode.
    The program will continue to ask its user as long as the user
    inputs integers. If the user inputs anything other than integers[i.e. any character like- 'a', etc],
    then the program will terminate, and end its process. But if the user inputs any decimal number,
    the program will only count the integer part of the input.
```

Coder : Mosfiqur Rahman (mr986@drexel.edu)

Last Modified : 9th February, 2016.

```
*****
*****/
```

Program Abstract:

The name of the program is postal_bar_code_converter. It's written in C++. The program is created to convert a five digit zipcode into a proper barcode. It will be very easy for its user to use because after running the program, the program will ask the user to enter the five digit zipcode as an input. Then program will show the calculated barcode instantly. The user will need to enter a non-negative integer as an input. If the user gives a negative number as an input, the program will show an error message, and will let the user type his/her zipcode correctly again. However, if the user types a decimal number, the program will just count the integer part before the decimal point, and if the user types a character as an input then the program will terminate, and end its process.

Basic Functions:

To make the program work successfully, 3 useful functions is used.

Function 1: "int makeCheckDigit (int zipcode)"

This function will find the check-digit for a zipcode.

[Check-digit:- A check-digit is the value of the summation of all digits of a number, and it's chosen to make the sum a multiple of 10.]

Here, 'zipcode', an integer variable, is the only parameter which will be taken by this function.

After passing the value of the parameter, the function will go through a few steps-

step 1:- It will run the value through a while loop to find the sum of the digits, and will continue calculating as long as its done with the following sub-steps.

sub-step 1:- To find the sum, it will first find the modulus 10 of the passed value.

sub-step 2:- Then it will store the value as an integer value in the variable 'sum_of_digits'.

sub-step 3:- After that, it will divide the value of the variable 'zipcode' by 10, and will renew the integer value of the variable zipcode.

sub-step 4:- The program will run sub-step 1 to 3 as long as it is done with the last digit of the

value of

'zipcode'.

step 2:- Now to find the desired check digit, the function will subtract the value of the modulus 10 of the value of the

variable 'sum_of_digits', and will store the value in an integer variable check_digit.

At last, it will return the check digit value of the zipcode, which was stored in the variable check_digit after passing the value of zipcode.

Function 2: "string convertDigit(int value)"

This function will convert individual digits of the zipcode to its individual barcodes. If someone gives more than one digit as an input, it will generate an alert message of "invalid" output.

Here, 'value', an integer variable is the only parameter, which will be taken by this function.

After passing the value of the parameter, it will pass the value through some logical statements[if-else statements], and check what the digit is.

Again, there are different return values[i.e. barcode values] for every digit from 0 to 9.

However, if anything else than a digit from 0 to 9 is passed, it will also generate an alert message of "invalid" output.

At last, it will return the barcode value of an individual digit after passing the value of the variable 'value'.

Function 3: "string barcode(int zipcode)"

This function will convert the whole zipcode into its complete barcode.

Here, 'zipcode', an integer variable, is the only parameter, which will be taken by the function.

After passing the value of the parameter, it will call the 'makeCheckDigit' function, and pass the value again to this function.

By using the 'makeCheckDigit' function, it will get the check digit of the passed value, and will store the value in an integer variable check_digit.

Now it stores the barcode value of every single digit of the zipcode by placing one after another in a string variable 'prefix' starting with "|" at the front using the convertDigit function.

Here, to separate the digits from the whole zipcode,

step-1:- the function will find the modulus 10 of the zipcode, and store the value in a integer variable digit_1.

step-2:- Then it'll divide the value of zipcode by 10 and renew the value in the variable zipcode.

step-3:- the function will follow step 1 and 2 till it's done with the last digit, store the values in variables digit_2, digit_3, digit_4, digit_5 respectively as there are five digits in a zipcode.

Now, the barcode function will pass the values of the variables from digit_5 to digit_1 in the convertDigit function, and will store the complete value in the string 'prefix'.

This is the first part of the complete barcode of the passed zipcode.

To get the rest of the part of the barcode, the barcode function will call the convertDigit function again, and will find the barcode for the check digit.

After that, it will store the value in another string 'postfix', placing a "|" after that.

At last, it will return the value of the string prefix and postfix all together.

Main Program Functionality:

Here, in the written program, as the above three functions are defined after the main function, their function prototype is stated properly before the main function.

Inside the main function, there are two integer variables `zipcode`, and `check_digit`; and a string variable `bar_code_value`. The variable `zipcode` is to store the input value of `zipcode` from the user.

Then the `check_digit` variable is created to store the value of the check digit of the `zipcode`, and the string variable `bar_code_value` is used to store the complete value of a converted bar code from given input[i.e. `zipcode`].

This function first prints out some introductory texts about the whole program, and then it runs a while loop to get the input of the `zipcode` from the user. After getting the input from the user,

it will first check whether it's positive or negative through a simple if-else statement. Now, if the input is negative it prints out an error message, and lets the user give his/her input again.

However, if the input is positive, then it works as follows:

1. It calls the `makeCheckDigit` function to find the check digit of the `zipcode` by passing the value of `zipcode` to the function. Then it stores the value in the `check_digit` variable, and prints out the value of the check digit.
2. After that, it calls the `barcode` function to find the complete barcode of the given input.[i.e. `zipcode`] by passing the value of `zipcode` to the function. Then it stores the value in the variable `bar_code_value`, and prints out the complete barcode for the given input.[i.e. `zipcode`].

