

Expressions

Workshop 2 (worth 3% of your final grade)

URL: <https://github.com/Seneca-144100/IPC-WS2>

In this workshop, you are to write a program that receives the number of, Toonies, Loonies, Quarters, Dimes, Nickels and Pennies in a piggy bank and calculates the total amount of money (balance) saved in the piggy bank in Dollars. Then the program should breakdown the balance to least amount of bills and coins.

LEARNING OUTCOMES

Upon successful completion of this workshop, you will have demonstrated the abilities to:

- do simple calculation using C operators and expressions.
- use scanf to read a number from console
- cast values between integer and floating point

SUBMISSION POLICY

Your workshops are divided in two sections; [in_lab](#) and [at_home](#).

The “[in_lab](#)” section is to be completed **during your assigned lab section**. It is to be completed and submitted by the end of the workshop. If you do not attend the workshop, you can submit the “[in_lab](#)” section along with your “[at_home](#)” section (a 21% late deduction will be assessed). The “[at_home](#)” portion of the lab is **due the day before your next scheduled workshop**

All your work (all the files you create or modify) must contain your name, Seneca email and student number.

You are responsible for regularly backing up your work.

IN-LAB: ITEM CLASS (70%)

Download or clone workshop 2 from <https://github.com/Seneca-144100/IPC-WS2>

In the `in_lab` directory of workshop 2 click on `in_lab.vsxproj` to open the workshop in Visual Studio.

In solution explorer open and write your code in piggybank.c for workshop 2.

Start the program by asking the user to enter the number of coins. Print the following message:

Please enter the number of coins in your piggy bank: (go to newline)

Then print the name of the coins one by one and get the number of the coins after each name.

Assuming the user entering "9" for all of them, it should work like this:

(**<ENTER>** means hitting the **ENTER** key)

```
Toonies: 9<ENTER>
Loonies: 9<ENTER>
Quarters: 9<ENTER>
Dimes: 9<ENTER>
Nickels: 9<ENTER>
Pennies: 9<ENTER>
```

Please note that a space is printed after ":".

After receiving the number of Pennies calculate the total value of the money in the Piggybank:

Multiply the number of each coin to its Dollar value (for example Toonie has the value 2 and Quarter has the value 0.25) and store the sum in a floating point variable.

$$\begin{aligned} \text{Total Amount} = & \text{Toonies} \times 2 + \text{Loonies} + \\ & \text{Quarters} \times 0.25 + \text{Dimes} \times 0.10 + \\ & \text{Nickels} \times 0.05 + \text{Pennies} \times 0.01 \end{aligned}$$

Then print the total amount using this message:

You have \$119.79 in your piggy bank!(go to newline)

Replace **\$119.79** with the actual value calculated for total in you program.
Note that there are only two digits after the decimal point.

Execution and Output Example:

```
Please enter the number of coins in your piggy bank:
Toonies: 20
Loonies: 10
Quarters: 5
Dimes: 3
Nickels: 2
```

Pennies: 1
You have \$51.66 in your piggy bank!

For submission instructions, see the [SUBMISSION](#) section below.

IN_LAB SUBMISSION:

To test and demonstrate execution of your program using the same data as the output example above.

If not on matrix already, upload your [piggybank.c](#) to your matrix account. Compile and run your code and make sure everything works properly.

```
AtThePrompt> gcc -Wall piggybank.c
```

-Wall activates the display of warnings in GCC compiler.

Then run the following script from your account: (replace profname.proflastname with your professors Seneca userid)

```
~profname.proflastname/submit ipc_w2_in_lab <ENTER>
```

and follow the instructions.

AT_HOME: (20%)

After completing the [in_lab](#) section, edit and upgrade [piggybank.c](#) to breakdown the total balance to least amount of bills and coins, assuming our program does not use bills greater than 50 Dollars.

For example if the total amount calculated is \$98.37, the program should break this value to following:

\$50 bill: 1
\$20 bill: 2
\$10 bill: 0
\$5 bill: 1
Change: \$3.37

To do this create first create integer variables to hold the number of bills and coins and also a floating point variable to hold the change.

Then separate the integer and fractional part of the calculated total (in this example 98 and 0.37). Let's call the integer part iTotal and keep the fractional part in the "change" variable.

To find the number of bills do the following:

- 1- Start from largest bill (50\$ bill).
- 2- Divide iTotal to the value of the bill and keep it in the corresponding variable for that bill.
- 3- Reduce iTotal by the value of the bills set aside.
- 4- Repeat 2 and 3 for the next bill value until you get to five dollar bill.
- 5- Add whatever that is left to the value of change and print the values.

Example:

Number of 50 Dollar bills = $98 / 50$ that is 1.

Reduce 98 by (50×1) that is 48.

Number of 20 Dollar bills = $48 / 20$ that is 2;

Reduce 48 by (20×2) that is 8

Number of 10 Dollar bills = $8 / 10$ that is 0;

Reduce 8 by (0×10) that is 0.

Number of 5 Dollar bills = $8 / 5$ that is 1;

Total of 3 dollars is left.

Add 3 to the change and it becomes \$3.37

Output Example:

Please enter the number of coins in your piggy bank:

Toonies: 45

Loonies: 8

Quarters: 1

Dimes: 0

Nickels: 2

Pennies: 2

You have \$98.37 in your piggy bank!

\$50 bill: 1

\$20 bill: 2

\$10 bill: 0

\$5 bill: 1

Change: \$3.37

AT-HOME REFLECTION (10%)

Please provide brief answers to the following questions in a text file named `reflect.txt`.

1- What did you learn in this workshop?

AT_HOME SUBMISSION:

To test and demonstrate execution of your program using the same data as the output example above.

If not on matrix already, upload your `piggybank.c` to your matrix account. Compile and run your code and make sure everything works properly.

```
AtThePrompt> gcc -Wall piggybank.c
```

Then run the following script from your account: (replace profname.proflastname with your professors Seneca userid)

```
~profname.proflastname/submit ipc_w2_at_home <ENTER>
```

and follow the instructions.