Coding Assignment 3

1. Create `Code3_xxxxxxxxxx.cpp`

Copy your working version of `Code2_xxxxxxxxxx.cpp` to `Code3_xxxxxxxxxx.cpp`.

2. Create `SnackMachine.cpp`

Move the member function code out of the `SnackMachine`'s class structure. Use the scope resolution operator to tie the prototypes left in the class definition to the actual member function code in `SnackMachine.cpp`. Only the function prototypes and data members should still reside in `SnackMachine.h`.

3. Add `include` guard to `SnackMachine.h`

Add an `include` guard to `SnackMachine.h`. Use the name `SNACK_MACHINE_H` in your `include` guard.

4. `SnackLib.h`

Download `SnackLib.cpp` and `SnackLib.h` from Canvas. **Do not alter these files.** If you alter them to make your code work, then your code will fail when graded because the graders will not use your versions – they will be using the versions from Canvas to grade your assignment. Be sure to add `include`s where needed to use the new function in `SnackLib`.

5. `makefile`

Create a new makefile that can compile these files and creates an executable named `Code3_xxxxxxxxxx.e`

`Code3_xxxxxxxxxx.cpp`

`SnackMachine.cpp`

`SnackLib.cpp`

`SnackLib.h`

`SnackMachine.h`

6. Add two new member functions – `setMachineName()` and `setSnackPrice()`

Add a new member function called `setMachineName()`. Add it as menu item 5, "`Update Machine Name`". Prompt the user "`Enter a new machine name`". Take in `newMachineName` and call `setMachineName()` with one `string` parameter of `newMachineName` and no return value. After calling member function, print "`Machine name has been updated`". If your option 4 to display the machine info does not include showing the machine name, then add calling `getMachineName()` and printing out the machine name. Be sure to accept names containing spaces (use `getline` rather than `cin`).

Add a new member function called `setSnackPrice()`. Add it as menu item 6, "`Update Snack Price`". Prompt the user "`Enter a new Snack price`". Take in `newSnackPrice` and call `setSnackPrice()` with one `int` parameter of `newSnackPrice` and no return value. After calling member function, print "`Snack price has been updated`".

7. Add default parameter to `SnackMachine`'s constructor

Update `SnackMachine`'s constructor to take defaults for all parameters.

Default Values – `machineName` of "New Machine", Snack price of 50, change level of 500 and inventory level of 100.

8. Command line parameters

Add a new function, `get_command_line_params` to `Code3_xxxxxxxxxx.cpp` that contains ALL of the following code

Return type

`void`

Parameters

`argv` and `argc`

string to hold and pass back input file name

string to hold and pass back output file name

You should look for an input file and an output file – the input file will be listed first and the output file second.

Using `argc`, determine if no command line parameters were passed and, if none were, then throw an exception of `invalid_argument` with a string of "Missing command line parameters - - Usage : input^filename output^filename"

If either command line argument is missing, then throw an exception of `invalid_argument` with a string of "Missing command line parameters - - Usage : input^filename output^filename"

Your program will be run using

`./Code3_xxxxxxxxxx.e xxxxx yyyyy`

where `xxxxx` is the input file and `yyyyy` is the output file.

9. Overload the << operator

Overload << to print out the machine information that menu option 4 "Display Machine Info" currently calls member functions to do. Menu option 4 in `Code3_xxxxxxxxxx.cpp` should now be one `cout` statement and a `break`.

10. Code3_xxxxxxxxxx.cpp

`main()` should take in `argv` and `argc`.

Use a try-catch block inside `get_command_line_params()` to take action if a parameter was missing. If function throws an exception, then catch it, print the string from the exception object and exit from the program to immediately terminate.

Create a `vector` of type `SnackMachine` called `SetOfSnackMachines`.

Open the input file using file handle `SnackInputFile`.

Open the output file using file handle `SnackOutputFile`. Open the file with `ios::out`.

If `SnackInputFile` is open

While `getline()` reads a line from `SnackInputFile` into `SnackMachineLine` (`string` variable)

Call `ParseSnackLine()` from `SnackLib.cpp` with two parameters

`SnackMachineLine`

`ConstructorValues` - a `vector` of type `string` of size 4

If `ParseSnackLine()` returns TRUE

Using each of the elements returned in `ConstructorValues`, construct a temporary `SnackMachine` object.

Push that temporary object into your `SetOfSnackMachine vector`.

Else (`SnackInputFile` did not open)

print "`Unable to open file`" and `exit()`

do-while machine picked is not the exit option of 0

Ask user "`Pick a Snack Machine`"

Use a `for` loop to iterate from `0` to `size()` of `SetOfSnackMachines`

For each `SnackMachine` object, print the loop counter and the object's machine name. Make option 0 the exit. You will need to deal with offsetting the vector index with the menu index in order to use 0 as the exit option. Print out a final option to add a new machine.

```
Pick a Snack Machine

0. Exit

1. Machine Bugs Bunny

2. Machine Cecil Turtle

3. Machine Daffy Duck

4. Machine Elmer Fudd

5. Machine Fog Horn

6. Add a new machine

Enter choice
```

```
Machine Bugs Bunny|50|500|50
Machine Cecil Turtle|45|545|45
Machine Daffy Duck|40|540|1
Machine Elmer Fudd|100|1000|10
Machine Fog Horn|35|350|99
```

If choice is 0,

then write all Snack Machines to the output file using the same pipe delimited format and order of fields as the input file and use `return` to end program. Each pipe delimited line of file output should be created by calling function `CreateSnackOutputLine`. Pass each object from `SetOfMachines` and `CreateSnackOutputLine()` will directly access the necessary private data members to form the pipe delimited output line. `CreateSnackOutputLine()` is not a member function. The output file created should be able to serve as an input file for the next run of your program.

If choice is to add a new machine, call the default constructor and add the new machine to the `SetOfSnackMachines vector` and print "`New machine added`".

`do-while` user wants to manipulate a single Snack Machine (same `do-while` already in code)

Display existing menu of options for a single Snack Machines (same menu from previous assignment)

Take choice from menu and allow operations on chosen machine - same code from previous assignment except that it must use the chosen Snack Machine from the "`Pick a Snack Machine`"menu. Remember that if you added a new machine,then that is the machine being manipulated currently.

Sample Output

```
student@cse1325:/media/sf_VM/CA3$ ./Code3_1000074079.e InputSnackFile.txt Output.txt
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. Add a new machine

Enter choice 0
```

```
student@cse1325:/media/sf_VM/CA3$ ./Code3_1000074079.e InputSnackFile.txt Output.txt
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. Add a new machine

Enter choice 1



Machine Bugs Bunny



0. Walk away
1. Buy a Snack
2. Restock Machine
```

```
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 0
Are you sure you aren't really HUNGRY and need a Snack?
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. Add a new machine

Enter choice 0
```

```
student@cse1325:/media/sf_VM/CA3$ ./Code3_1000074079.e InputSnackFile.txt Output.txt
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. Add a new machine

Enter choice 1



Machine Bugs Bunny
```

```
0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 4

Machine Bugs Bunny

Current Inventory Level 50
Max Inventory Capacity  100

Current Change Level    $5.00
Max Change Capacity     5000

Current Snack price is  $0.50


0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 6
Enter a new Snack price 51
Snack price has been updated
```

```
0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 4

Machine Bugs Bunny

Current Inventory Level 50
Max Inventory Capacity  100

Current Change Level    $5.00
Max Change Capacity     5000

Current Snack price is  $0.51


0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 0
Are you sure you aren't really HUNGRY and need a Snack?
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
```

```
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. Add a new machine

Enter choice 1



Machine Bugs Bunny



0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 4

Machine Bugs Bunny

Current Inventory Level 50
Max Inventory Capacity  100

Current Change Level    $5.00
Max Change Capacity     5000

Current Snack price is  $0.51



0. Walk away
1. Buy a Snack
```

```
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 0
Are you sure you aren't really HUNGRY and need a Snack?
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. Add a new machine

Enter choice 6
New machine added


Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. New Machine
7. Add a new machine

Enter choice 6
```

New Machine

0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 5
Enter a new machine name Machine Road Runner

0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 4

Machine Road Runner

Current Inventory Level 100
Max Inventory Capacity 100

Current Change Level $5.00
Max Change Capacity   5000

Current Snack price is $0.50

0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 0
Are you sure you aren't really HUNGRY and need a Snack?
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. Machine Road Runner
7. Add a new machine

Enter choice 3

Machine Daffy Duck

0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name

```
6. Update Snack Price

Choice : 1
A snack costs $0.40

Insert payment 43


Here's your Snack and your change of $0.03


0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 4

Machine Daffy Duck

Current Inventory Level 0
Max Inventory Capacity  100

Current Change Level    $5.80
Max Change Capacity     5000

Current Snack price is  $0.40


0. Walk away
1. Buy a Snack
2. Restock Machine
```

```
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 1
Out of snacks - call 1800RESTOCK to request a restock...returning your payment



0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 2
How much product are you adding to the machine?  45


Your machine has been restocked


Your inventory level is now 45


0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price
```

```
Choice : 4

Machine Daffy Duck

Current Inventory Level 45
Max Inventory Capacity  100

Current Change Level    $5.80
Max Change Capacity     5000

Current Snack price is  $0.40



0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 0
Are you sure you aren't really HUNGRY and need a Snack?
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. Machine Road Runner
7. Add a new machine
```

```
Enter choice 0
```

```
student@cse1325:/media/sf_VM/CA3$ more Output.txt
Machine Bugs Bunny|51|500|50
Machine Cecil Turtle|45|545|45
Machine Daffy Duck|40|580|45
Machine Elmer Fudd|100|1000|10
Machine Fog Horn|35|350|99
Machine Road Runner|50|500|100
```

```
student@cse1325:/media/sf_VM/CA3$ ./Code3_1000074079.e Output.txt NewOutput.txt
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. Machine Road Runner
7. Add a new machine

Enter choice 6



Machine Road Runner



0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price
```

```
Choice : 4

Machine Road Runner

Current Inventory Level 100
Max Inventory Capacity  100

Current Change Level    $5.00
Max Change Capacity     5000

Current Snack price is  $0.50



0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 0
Are you sure you aren't really HUNGRY and need a Snack?
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
6. Machine Road Runner
7. Add a new machine
```

```
Enter choice 1


Machine Bugs Bunny


0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 1
A snack costs $0.51

Insert payment 50
Insufficent payment...returning your payment



0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 6
Enter a new Snack price 32
Snack price has been updated
```

```
0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 1
A snack costs $0.32

Insert payment 51


Here's your Snack and your change of $0.19


0. Walk away
1. Buy a Snack
2. Restock Machine
3. Add change
4. Display Machine Info
5. Update Machine Name
6. Update Snack Price

Choice : 0
Are you sure you aren't really HUNGRY and need a Snack?
Pick a Snack Machine

0. Exit
1. Machine Bugs Bunny
2. Machine Cecil Turtle
3. Machine Daffy Duck
4. Machine Elmer Fudd
5. Machine Fog Horn
```

```
6. Machine Road Runner
7. Add a new machine

Enter choice 0
```

```
student@cse1325:/media/sf_VM/CA3$ more Output.txt
Machine Bugs Bunny|51|500|50
Machine Cecil Turtle|45|545|45
Machine Daffy Duck|40|580|45
Machine Elmer Fudd|100|1000|10
Machine Fog Horn|35|350|99
Machine Road Runner|50|500|100
```

```
student@cse1325:/media/sf_VM/CA3$ more NewOutput.txt
Machine Bugs Bunny|32|532|49
Machine Cecil Turtle|45|545|45
Machine Daffy Duck|40|580|45
Machine Elmer Fudd|100|1000|10
Machine Fog Horn|35|350|99
Machine Road Runner|50|500|100
```

```
student@cse1325:/media/sf_VM/CA3$ more InputSnackFile.txt
Machine Bugs Bunny|50|500|50
Machine Cecil Turtle|45|545|45
Machine Daffy Duck|40|540|1
Machine Elmer Fudd|100|1000|10
Machine Fog Horn|35|350|99
student@cse1325:/media/sf_VM/CA3$
```

```
student@cse1325:/media/sf_VM/CA3$ ./Code3_1000074079.e
Exiting program -

Missing command line parameters - Usage : INPUTFILENAME OUTPUTFILENAME
```

```
student@cse1325:/media/sf_VM/CA3$
```

```
student@cse1325:/media/sf_VM/CA3$ ./Code3_1000074079.e InputSnackFile.txt
Exiting program -

Missing command line parameters - Usage : INPUTFILENAME OUTPUTFILENAME

student@cse1325:/media/sf_VM/CA3$
```