```cpp
// Froyo Assignment
#include < iostream >
#include < string >
#include < sstream >
using namespace std;

int main()
{
    string response, numFroyo, usedStamps;
    int purchase, stampBalance;

    purchase = 0;
    stampBalance = 0;

    cout << "Welcome to Froyo Hill. Pleace choose from the following.\n\n";

    do
    {
        cout << "----- Menu -----\n\n"
            << "P (Process Purchase)\n"
            << "S (Shut Down)\n";
        getline(cin, response);

        if (response == "p" || response == "P")
        {

            if (stampBalance >= 10)
            {
                cout << "You have earned enough stamps for a free yogurt.\n"
                    << "Would you like to you like yo use your stamps?\n"
                    << "Yes or No\n";
                getline(cin, usedStamps);

                if (usedStamps == "y" || usedStamps == "Y")
                {
                    stampBalance -= 10;
                    cout << "You have used 10 stamps and have " << stampBalance << "
left.\n"
                        << "Please enjoy.\n\n";
                    continue;
                }
            }
            if (response == "p" || response == "P" || stampBalance > 10 || usedStam
ps == "N" || usedStamps == "n")
            {
                cout << "\nHow many yogurts would you like to buy?\n";
                getline(cin, numFroyo);
                istringstream(numFroyo) >> purchase;

                if (purchase > 0)
                {
                    stampBalance += purchase;
                    cout << "You have earned " << purchase << " stamps. \n"
                        << "You now have a stamp balance of " << stampBalance << ".\n
\n";
                }
                else if (purchase < 0)
                {
                    cout << "ERROR!! Invalid Input!\n\n";
                }
            }
        }
    }
```

Handwritten annotations:
- "off-spec, compares whole line instead of first char" (pointing to `if (response == "p" || response == "P")`)
- "magic #" (pointing to `10`)
- circled: `yo`, `10`, `10`, `left.\n`, `ps == "N"`, `balance`, `\n";`
- "overly complex condition" (pointing to the second if statement)
- boxed: `if (purchase < 0)`

```cpp
        else if (response == "s" || response == "S")
        {
            cout << "You have decided to not make a pruchase.\n"
                << "Good-bye.\n";
            break;
        }

        else if (response != "s" && response != "S")
        {
            cout << "ERROR!! Invalid Input!\n"
                << "Please enter P or S.\n\n";
        }

    } while (response != "S" && response != "s");
}
```

unnecessary because of break above

```
/*------------------------- Posted Run #1 ------------------------------

Welcome to Froyo Hill. Pleace choose from the following.

----- Menu -----

P (Process Purchase)
S (Shut Down)
p

How many yogurts would you like to buy?
2
You have earned 2 stamps.
You now have a stamp balanace of 2.

----- Menu -----

P (Process Purchase)
S (Shut Down)
p

How many yogurts would you like to buy?
4
You have earned 4 stamps.
You now have a stamp balanace of 6.

----- Menu -----

P (Process Purchase)
S (Shut Down)
p

How many yogurts would you like to buy?
10
You have earned 10 stamps.
You now have a stamp balanace of 16.

----- Menu -----

P (Process Purchase)
S (Shut Down)
p
You have earned enough stamps for a free yogurt.
```

```
Would you like to you like yo use your stamps?
Yes or No
n
How many yogurts would you like to buy?
2
You have earned 2 stamps.
You now have a stamp balanace of 18.

----- Menu -----

P (Process Purchase)
S (Shut Down)
p
You have earned enough stamps for a free yogurt.
Would you like to you like yo use your stamps?
Yes or No
y
You have used 10 stamps and have 8 left.
Please enjoy.

----- Menu -----

P (Process Purchase)
S (Shut Down)
p

How many yogurts would you like to buy?
-4
ERROR!! Invalid Input!

----- Menu -----

P (Process Purchase)
S (Shut Down)
r
ERROR!! Invalid Input!
Please enter P or S.

----- Menu -----

P (Process Purchase)
S (Shut Down)
s
You have decided to not make a pruchase.
Good-bye.
Press any key to continue . . .

---------------------------------------------------------------------
-*/
/*------------------------- Posted Run # 2-------------------------------
----

   Welcome to Froyo Hill. Pleace choose from the following.

----- Menu -----

P (Process Purchase)
S (Shut Down)
p

How many yogurts would you like to buy?
10
```

```
You have earned 10 stamps.
You now have a stamp balanace of 10.

----- Menu -----

P (Process Purchase)
S (Shut Down)
p
You have earned enough stamps for a free yogurt.
Would you like to you like yo use your stamps?
Yes or No
n

How many yogurts would you like to buy?
-1
ERROR!! Invalid Input!

----- Menu -----

P (Process Purchase)
S (Shut Down)
p
You have earned enough stamps for a free yogurt.
Would you like to you like yo use your stamps?
Yes or No
n

How many yogurts would you like to buy?
2
You have earned 2 stamps.
You now have a stamp balanace of 12.

----- Menu -----

P (Process Purchase)
S (Shut Down)
p
You have earned enough stamps for a free yogurt.
Would you like to you like yo use your stamps?
Yes or No
y
You have used 10 stamps and have 2 left.
Please enjoy.

----- Menu -----

P (Process Purchase)
S (Shut Down)
p

How many yogurts would you like to buy?
2
You have earned 2 stamps.
You now have a stamp balanace of 4.

----- Menu -----

P (Process Purchase)
S (Shut Down)
t
ERROR!! Invalid Input!
Please enter P or S.
```

```
----- Menu -----

P (Process Purchase)
S (Shut Down)
s
You have decided to not make a pruchase.
Good-bye.
Press any key to continue . . .

---------------------------------------------------------------------
_*/
```

```cpp
// CS 2A Lab 4
// Instructor Solution:
//  Original - Prof. Loceff, Updates, Edits, Annotations: &

// Notes:
//  - Use of sensible names for vars
//  - Correct use of global consts
//  - Use of symbolic consts rather than literals (magics)
//  - Use of correct data types (ints and doubles)
//  - Use of getline(cin) rather than cin >>
//  - Correct arithmetic
//  - Faithfulness to spec (reporting per serving, etc.)
//
// Observe how a careful use of break/continue statements saves a
// significant number of ELSEs and indentation. Also, by handling
// the choices in a particular sequence we can dismiss the non-trivial
// cases early and not have to duplicate code (e.g. the purchase logic)
//
#include <iostream>
#include <string>
#include <sstream>
using namespace std;

// main client -------------------------------------------------------
int main()
{
    const int STAMPS_PER_FREE_YOG = 10;

    string userStr;
    int userInt;
    int numStamps = 0;
    char letter;

    // Will break on input of "quit".
    while (true) {
        cout << "\nMenu: "<<
        "\n   P (process Purchase)"<<
        "\n   S (Shut down) \n\n   Your Choice: ";
        getline(cin, userStr);

        if (userStr.length() < 1) {
            cout << "\n *** Please enter a valid choice (P or S). ***\n";
            continue;
        }
        letter = toupper(userStr[0]);    // store in letter

        if (letter == 'S') {
            cout << "\nSystem shutting down.\n";
            break;
        }

        // Note - Omitted ELSE
        if (letter != 'P') {
```

```cpp
            cout << "\n *** Use P or S, please. ***\n";
            continue;
        }

        // At this point, you can be sure that they entered P - right?
        // So no need for redundant check if (letter == 'P')

        // Freebie check first - According to the spec
        if (numStamps >= STAMPS_PER_FREE_YOG) {
            cout << "\nYou qualify for a free yogurt."
                << "\nWould you like to use your credits? (Y or N) ";
            getline(cin, userStr);
            if (userStr.length() < 1 || toupper(userStr[0] != 'N')) {
                cout << "\n *** Invalid choice. Must be Y or N. ***\n";
                continue;
            }

            if (toupper(userStr[0]) == 'Y') {
                numStamps -= STAMPS_PER_FREE_YOG;
                cout << "\nYou have just used " <<  STAMPS_PER_FREE_YOG
                    <<  " credits and have " <<  numStamps <<  " left."
                    << "\nEnjoy your free yogurt.\n";
                continue;
            }
        } // Freebie check finished.

        // Note: At this point, either they aren't eligible for
        // a freebie OR they were and said N to using stamps.
        // So - A normal transaction.
        // You don't have to duplicate the purchase logic if you
        // factor your code like this - and it's more readable.

        cout << "\nHow many yogurts would you like to buy ? ";
        getline(cin, userStr);
        istringstream(userStr) >> userInt ;
        if (userInt < 1) {
            cout << "\n *** Invalid quantity. Must be at least 1. ***\n";
            continue;
        }

        numStamps += userInt;
        cout << "\n You just earned " <<  userInt <<  " stamps"
            <<  " and have a total of "<<  numStamps
            <<  " to use.\n";
    } // main loop

    cout << "\nGoodbye.\n\n";
}

/* ------------------------ run ---------------------



 Menu:
```

```
P (process Purchase)
S (Shut down)

Your Choice: a

*** Use P or S, please. ***

Menu:
P (process Purchase)
S (Shut down)

Your Choice: P

How many yogurts would you like to buy ? 0

*** Invalid quantity. Must be at least 1. ***

Menu:
P (process Purchase)
S (Shut down)

Your Choice: P

How many yogurts would you like to buy ? 1

You just earned 1 stamps and have a total of 1 to use.

Menu:
P (process Purchase)
S (Shut down)

Your Choice: P

How many yogurts would you like to buy ? 9

You just earned 9 stamps and have a total of 10 to use.

Menu:
P (process Purchase)
S (Shut down)

Your Choice: p

You qualify for a free yogurt.
Would you like to use your credits? (Y or N) n

How many yogurts would you like to buy ? 0

*** Invalid quantity. Must be at least 1. ***

Menu:
P (process Purchase)
S (Shut down)
```

```
Your Choice: P

You qualify for a free yogurt.
Would you like to use your credits? (Y or N) 20

*** Invalid response. Must be Y or N. ***

Menu:
P (process Purchase)
S (Shut down)

Your Choice: P

You qualify for a free yogurt.
Would you like to use your credits? (Y or N) y

You have just used 10 credits and have 0 left.
Enjoy your free yogurt.

Menu:
P (process Purchase)
S (Shut down)

Your Choice: P

How many yogurts would you like to buy ? 20

You just earned 20 stamps and have a total of 20 to use.

Menu:
P (process Purchase)
S (Shut down)

Your Choice: p

You qualify for a free yogurt.
Would you like to use your credits? (Y or N) y

You have just used 10 credits and have 10 left.
Enjoy your free yogurt.

Menu:
P (process Purchase)
S (Shut down)

Your Choice: p

You qualify for a free yogurt.
Would you like to use your credits? (Y or N) y

You have just used 10 credits and have 0 left.
Enjoy your free yogurt.
```

```
Menu:
P (process Purchase)
S (Shut down)

Your Choice: s

System shutting down.

Goodbye.

Program ended with exit code: 0

-------------------------------------------------------------- */
```