

# Project 1

<CSC 5 CAFÉ>

**CSC 5 – 46091**

**Name: Nguyen, Johny**

**Date: 07/20/15**

**Introduction:**

Title: CSC 5 Café

This game revolves around the idea of a restaurant that requires the user to guess what's on the menu. If the user is able to successfully guess what's on the menu, then they'll be served at the restaurant.

The beginning of the game will require the user to input their name as a form of reservation for the restaurant. From there on, the game will prompt the user to select a level of difficulty from easy, average, and hard.

Within the game, the user will be allowed five tries to guess the appropriate letters for the hidden word.

***Example:***

*/\*Welcome to the CSC 5 Cafe! Guess what's on our menu today!*

At the CSC 5 Cafe, you will have to guess what's on our menu in order to be served!

Customer's First Name: Johny

Choose a LEVEL (Easy (1), Average (2), and Hard (3)):

Type in corresponding number for LEVEL of difficulty!

Input Level: Easy \*/

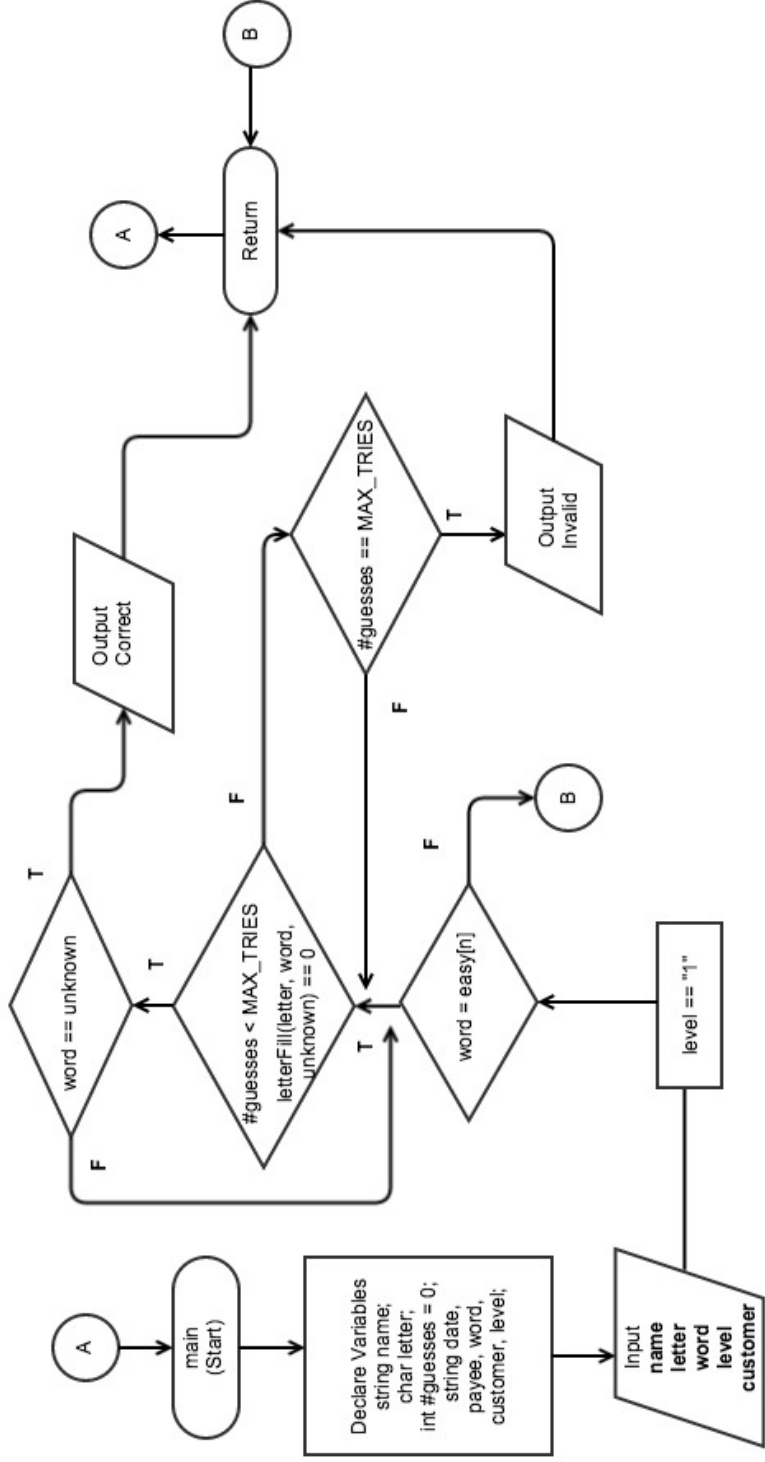
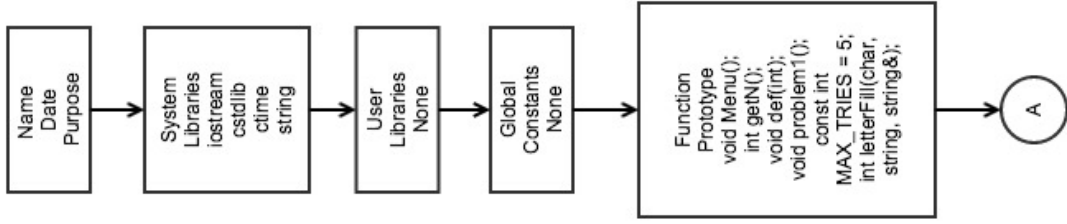
This game will enable individuals to guess the food that they'll be purchasing, as well as enjoy the experience!

**Summary/Description:**

Project size: 257 lines

Throughout this project, many concepts were utilized from what we have learned from the previous chapters inside the book. This project requires the use of various libraries, as well as functions that help user understand how to solve each specific word. From this game design, it took almost a whole week for me to work out the errors and how to incorporate the different levels of difficulties. Each level consist of a greater amount of hidden letters, but the same amount of tries. Although this game may seem simple, it will force the user to critically think about the words that will be used and how to strategically solve for each letters.

**Flow Chart:**



## CSC 5 PROJECT 1 : CAFE

### Major Variables Used:

Major Variables			
Type	Variable Name	Description	Location
Integer	getN()	Choose game function	Void Menu ()
	MAX_TRIES	Amount of tries	Int Main()
	letterFill	Asterisk letter fill	Int Main()
	#guesses	guesses left record	Int Main()
	matches	The matches with word	Int Main()
String	name	The user name	Void StartG()
	word	The random word	Void StartG()
	customer	Reference user	Void StartG()
	level	The input of difficulty	Void StartG()
	easy	The level	Void StartG()
	average	The level	Void StartG()
	hard	The level	Void StartG()
	nknown(word.length())	Unknown word key	Void StartG()

### C++ Constructs:

Chapters	New Syntax and Keywords	Location
Chapter 1	Equality operators and relational operators (==, !=, >, =, <=)	if (word == unknown);
	Arithmetic operators (+, -, *, /)	MAX_TRIES - #guesses
	Punctuation (, & ;)	throughout code
Chapter 2	Special Characters (   , #, <, >, {, }, ")	throughout code
	The "cout" Objects	cout<<"\nCustomer's First Name: ";
	The "endl;" Manipulator	cout << "\nWelcome to the CSC 5 Cafe! Guess what's on our menu today!" << endl;
	The escape "\n" Sequence	cout << "\n\n" << unknown\n;
	The "include" Directive	#include <iostream>
	Variable and Literals (int ; )	int getN() {
	The identifier (if, void, return, const, float, switch, char, case, break)	void game () {
	Integer Data Type (short, long, float)	within code
Chapter 3	The "cin" Object	cin>>customer;
	Type Casting	throughout code
	Constant and Array Sizes	string easy[] = { "bread", "bacon", "fruit", "grits" };

	Member Functions, such as "ignore"	cin.ignore();
Chapter 4	If statements	if (level == "1"){
	If/Else statements	else if (level == "2"){
	Menu- driven program organizer	void Menu();
	The conditional operator	if (letterFill(letter, word, unknown) == 0)
	Switch and break statements	break;
Chapter 5	The for loop	within code
	The increment and decrement operator (++ , __)	guesses++;
	The while loop	while (#guesses < MAX_TRIES)

### **Pseudo Code:**

*Initialize Program*

*If the menu screen appear*

*Display choices to start game*

*Input and enter a game number*

*If the level of choice is picked*

*The game will start and require the guessing of letters*

*If the guess is wrong, then the user will lose tries*

*Else if the option is average for difficulty*

*The game will start and require the guessing of letters*

*If the guess is wrong, then the user will lose tries*

*Else if the option is hard for difficulty*

*The game will start and require the guessing of letters*

*If the guess is wrong, then the user will lose tries*

*Once tries is exhaust, user will lose*

*If user loses the game, user can restart and play again*

*User can play multiple times and guess different words*

*Each level of difficulty presents more words and limit amounts of tries*

*If user guest letters and words correctly*

*The score will be set and user will try more difficult words*

*Print the final word and complete the game*

*Else*

*Start game again*

## References:

1. Cplusplus.com - The C++ Resources Network." *Cplusplus.com - The C++ Resources Network*. N.p., n.d. Web. 21 July 2015.
2. Gaddis, Tony. *Starting out with C*. Boston: Pearson Addison-Wesley, 2012. Print.

## Program Code:

**\* Author: Johny Man Nguyen**  
**\* Project: CSC Cafe Game**

### //System Libraries

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

### //Global Constants

### //Function Prototypes

```
void Menu();
int getN();
void def(int);
void problem1();
const int MAX_TRIES = 5;
int letterFill(char, string, string&);
```

### //Execution begins here

```
int main(int argv, char *argc[]){
    int inN;
    do{
        Menu();
        inN=getN();
        switch(inN){
            case 1: problem1();break;
            default:;
        };
    }while(inN<7);
    return 0;}
```

### //Menu Function

```
void Menu(){
    cout<<"Type in 1 and Enter to Start Game"<<endl;

    cout<<"\nType in 2 and Enter 2 to Exit Game\n"<<endl;
```

```

}
//Choose problem number function
int getN(){
    int inN;
    cin>>inN;
    return inN;
}

//Start Game
void game (){
    srand(time(NULL));
    string name;
    char letter;
    int num_of_wrong_guesses = 0;
    string word;

// Welcome the user
    cout << "\nWelcome to the CSC 5 Cafe! Guess what's on our menu today!" << endl;
    cout << "\nAt the CSC 5 Cafe, you will have to guess what's on our menu in order to be
served!" <<endl;
    string date, payee, customer;
    cout<<"\nCustomer's First Name: ";
    cin>>customer;

// Ask user for Easy, Average, Hard
    string level;
    cout << "\nChoose a LEVEL (Easy (1), Average (2), Hard (3)):" << endl;
    cout << "\nType in corresponding number for LEVEL of difficulty!"<<endl;
    cin >> level;

// Compare level
    if (level == "1"){
//Put all the string inside the array here
        string easy[] = { "bread", "bacon", "fruit", "grits" };
        string word;

        int n = rand() % 4;
        word = easy[n];

        //call the function here for guessing game
        // Initialize the secret word with the * character.
        string unknown(word.length(), '*');
        cout << "\n\nEach letter is represented by an asterisk.";
        cout << "\n\nYou have to type only one letter in one try.";
        cout << "\n\nYou have " << MAX_TRIES << " tries to try and guess the
food item.";
    }
}

```

```

        cout << "\n~~~~~";

// Loop until the guesses are used up
        while (num_of_wrong_guesses < MAX_TRIES)
        {
            cout << "\n\n" << unknown;
            cout << "\n\nGuess a letter: ";
            cin >> letter;

// Fill secret word with letter if the guess is correct,
// Otherwise increment the number of wrong guesses.
            if (letterFill(letter, word, unknown) == 0)
            {
                cout << endl << "Oops! That letter isn't in there!" << endl;
                num_of_wrong_guesses++;
            }
            else
            {
                cout << endl << "You found a letter! You're getting closer!" << endl;
            }

// Tell user how many guesses has left.
            cout << "You have " << MAX_TRIES - num_of_wrong_guesses;
            cout << " guesses left." << endl;

// Check if user guessed the word.
            if (word == unknown)
            {
                cout << word << endl;
                cout << "Yeah! You got it! (Press Enter to Play Again)";
                break;
            }
            if (num_of_wrong_guesses == MAX_TRIES)
            {
                cout << "\nSorry, you lose...we refuse to serve you! (Press Enter to Play Again)" << endl;
                cout << "\nThe word was : " << word << endl;
            }
            cin.ignore();
            cin.get();
        }

else if (level == "2"){
// Put all the string inside the array here
        string average[] = { "roastedham", "fruitsalad", "turkeywrap" };

        int n = rand() % 3;
        word = average[n];

```



```

// Call the function here for guessing game
// Initialize the secret word with the * character.
    string unknown(word.length(), '*');
    cout << "\n\nEach letter is represented by an asterisk.";
    cout << "\n\nYou have to type only one letter in one try.";
    cout << "\n\nYou have " << MAX_TRIES << " tries to try and guess the
    food item.";
    cout << "\n~~~~~";

// Loop until the guesses are used up
    while (num_of_wrong_guesses < MAX_TRIES)
    {
        cout << "\n\n" << unknown;
        cout << "\n\nGuess a letter: ";
        cin >> letter;

// Fill secret word with letter if the guess is correct,
// Otherwise increment the number of wrong guesses.
        if (letterFill(letter, word, unknown) == 0)
        {
            cout << endl << "Oops! That letter isn't in there!" << endl;
            num_of_wrong_guesses++; }
        else
        {
            cout << endl << "You found a letter! You're getting closer!" << endl;
        }
    }

// Tell user how many guesses have left.
    cout << "You have " << MAX_TRIES - num_of_wrong_guesses;
    cout << " guesses left." << endl;

// Check if user guessed the word.
    if (word == unknown)
    {
        cout << word << endl;
        cout << "Yeah! You got it! (Press Enter to Play Again)";
        break;
    }

    if (num_of_wrong_guesses == MAX_TRIES)
    {
        cout << "\nSorry, you lose...we refuse to serve you! (Press Enter to Play
        Again)" << endl;
        cout << "\nThe word was : " << word << endl;
    }
    cin.ignore();
    cin.get();
}

else if (level == "3"){

```

**//Put all the string inside the array here**

```
string hard[] = { "chickenburger", "frenchcrepes", "grilledsteak" };
```

```
int n = rand() % 3;
```

```
word = hard[n];
```

**// Call the function here for guessing game**

**// Initialize the secret word with the \* character.**

```
stringunknown(word.length(), '*');
```

```
cout << "\n\nEach letter is represented by an asterisk.";
```

```
cout << "\n\nYou have to type only one letter in one try.";
```

```
cout << "\n\nYou have " << MAX_TRIES << " tries to try and guess the  
food item.";
```

```
cout << "\n~~~~~";
```

**// Loop until the guesses are used up**

```
while (num_of_wrong_guesses < MAX_TRIES)
```

```
{
```

```
    cout << "\n\n" << unknown;
```

```
    cout << "\n\nGuess a letter: ";
```

```
    cin >> letter;
```

**// Fill secret word with letter if the guess is correct,**

**// Otherwise increment the number of wrong guesses.**

```
    if (letterFill(letter, word, unknown) == 0)
```

```
    {
```

```
        cout << endl << "Oops! That letter isn't in there!" << endl;
```

```
        num_of_wrong_guesses++;
```

```
    }
```

```
    else
```

```
    {
```

```
        cout << endl << "You found a letter! Isn't that exciting?" << endl;
```

```
    }
```

**// Tell user how many guesses has left.**

```
        cout << "You have " << MAX_TRIES - num_of_wrong_guesses;
```

```
        cout << " guesses left." << endl;
```

**// Check if user guessed the word.**

```
        if (word == unknown)
```

```
        {
```

```
            cout << word << endl;
```

```
            cout << "Yeah! You got it! (Press Enter to Play Again)" << endl;
```

```
            break;
```

```
        }
```

```
    }
```

```
    if (num_of_wrong_guesses == MAX_TRIES)
```

```
    {
```

```

        cout << "\nSorry, you lose...we refuse to serve you! (Press Enter to Play
        Again)" << endl;
        cout << "\nThe word was : " << word << endl;
    }
    cin.ignore();
    cin.get();
}
}

```

```

Int letterFill(char guess, string secretword, string &guessword)
{
    int i;
    int matches = 0;
    int len = secretword.length();
    for (i = 0; i < len; i++)
    {
        if (guess == guessword[i])
            return 0;
        if (guess == secretword[i])
        {
            guessword[i] = guess;
            matches++;
        }
    }
    return matches;
}

```

#### **//Exit Comment**

```

void def(int inN){
    cout<<"You typed "<<inN<<" to exit the program"<<endl;
}

```