Project 2

Title:
The Legend of
the
Amazingly Spectacular
Candy Bar

a text based RPG

Course:
CSC-5
Due Date:
7/31/15
Author:
Fred Roybal IV

Introduction

Title:The Legend of the Amazingly Spectacular Candy Bar

The Legend of the Amazingly Spectacular Candy Bar is a text based role-playing game where the user plays as a skilled explorer that is on the hunt for the Amazingly Spectacular Candy Bar.

The user will input commands based off of information provided by the game.

For example:

Menu**********
1: Attack
2: Use Skill
3: Aggressively Breakdance"
**********"

The game ends when either the main character is defeated in battle or reaches the end of the game.

Summary

Size of Project: 778lines when including spacing
Number of Functions: 7 (8 if you count main)
Number of Variables: approximately 24

The project contains the concepts featured in the Gaddis textbook up to chapter six and follows the checklist that was provided for this assignment.

The project itself was very fun to work on and doubly so when I was able to see the reactions of those who helped me test it out. Thanks to them, I was able to find any errors and fix them in a timely manner; much faster than if I had tried to find them myself.

Description

The driving force behind this project is to illustrate a comedic world that was inspired by other games like Candyland and movies like Indiana Jones. The simple user input I implemented allowed for the game to be easily enjoyed while still using the core concepts that were studies so far this semester.

Flow Chart

Pseudo Code

Initialize

Call Title Screen Function
Declare Variables

Prompt User to start the game and authenticate input via switch statement
    If input is a letter C, continue game
    Else exit program

Prompt User for input to generate main characters name
    if male,
        generate a male name
    else if female,
        generate a female name
    else if input is neither of the above
        generate a unisex alias

Output name to a file

Output text for Chapter 0-Tutorial

    Read name from file and output into text
    Continue to display text

Output text for Chapter 1-The Forest

    Call function for battle 1

    Initialize variables for both main character and enemy statistics
    Have enemy deal first attack
Do
    If main character health below 0
        return variable for Game Over
    If main character health above 0
        Prompt user for input on their turn
            If user chooses to use to attack
                execute basic attack code
            If user brings up skill menu
                execute a chosen skill
            Else if user inputs an invalid option
                skip player turn
    If enemy health is below 0
        return variable to continue game
    If enemy health is above 0
        enemy attacks player and deals damage
While the enemy is still up

Output text for Chapter 2- The Cave

Call function for battle 2

Initialize variables for both main character and enemy statistics
Call function to calculate poison damage
Do
If main character health below 0
return variable for Game Over
If main character health above 0
Prompt user for input on their turn
If user chooses to use to attack
execute basic attack code
If user brings up skill menu
execute a chosen skill
Else if user inputs an invalid option
skip player turn
If enemy health is below 0
return variable to continue game
If enemy health is above 0
enemy attacks player and deals damage
While the enemy is still up

If main character passes an event where a random number plus a variable is higher than a set amount,
Output a success message
Else
Output a failure message

Text continues on from this point

If main character passes another event where a random number plus a variable is higher than a set amount,
Output a success message
Else
Output a failure message

End of Program

Program

```
/*
 * File:   main.cpp
 * Author: Fred Roybal IV
 * Created on July 18, 2015, 3:10 PM
 * Purpose: Project 1. A text based RPG to show what was taught so far
 */

//System Libraries
#include <iostream>
#include <string>
#include <cstdlib>
#include <fstream>
#include <ctime>
using namespace std;

//User Libraries

//Global Constants
const unsigned int LIGHT=299792458;     //The speed of light in miles per sec.

//Function Prototypes

void TtlScrn ();                        //The title screen
void Continu ();                         //Used to pause text
unsigned short CritRt (short, float);       //Used to calculate critical rate
bool Battle1 (short, short, short, float);  //A battle with the first enemy

bool Battle2 (short, short, short, float,  //A battle with the second enemy
       short, short, short);
unsigned short Poison (short);              //used to calculate poison
short Stub(short);                          //Stub to test a function

//Execution Begins Here
int main(int argc, char** argv) {

   //Declare Variables
   srand(static_cast<unsigned short>(time(0)));//Sets random number seed

   char input[20];                          //Used for file input

   char txtName=0;                          //Used for file output

   const unsigned short ledge=10;             //Sets chances for event success

   bool Event1=false;                       //Used for distributing bonus
                                    //for event
```

```
bool gameOvr=false;                  //Used to end the game if player
                                     //is defeated

int NamSlct=0;                       //For choosing the main
                                     //characters name

struct nStrct{

string FName, MName, LName, FulName;       //Used for storing the name for
                                     //later use
};

nStrct name;

char contin;                         //Used to start the game from
                                     //The title screen

string MainC;                        //Used for selecting a name for
                                     //the main character



const short SIZE=4;

short mcStats[SIZE]={100,10,13,11};          //The main characters stats in
                                     //this order: max health,
                                     //strength, dexterity, and
                                     //magic

short cStats[SIZE]={100,10,13,11};           //Used for battle 2's enemy

float crit=10.5f;                    //Critical hit chance for
                                     //battle
//Begin Game

TtlScrn ();                          //Displays Title Screen

cout<<endl;
cout<<"Type (c) and press the enter key afterwards to continue"<<endl;
cout<<"Or type in a space or tab to stay here"<<endl;
cout<<"Or type anything else to exit"<<endl;

cin>>contin;                         //Used for waiting until the user
cout<<endl;                          //is ready

switch (contin){
   case 'C':
   case 'c': break;
```

```
    case 't': Stub(2);
    default: return 0;
}

//Generate Main Character's Name
cout<<"Before the game starts, are you male or female?"<<endl;
cout<<"This choice will generate the main character's name."<<endl;
cout<<"(Type 1 for male and 2 for female)"<<endl;
cin>>MainC;

if(MainC=="1"){

   name.FName="Joseph ";
   name.MName="Jerry ";
   name.LName="Jackson";

 }else if(MainC=="2"){

   name.FName="Bonnie ";
   name.MName="Bea ";
   name.LName="Boxley";

 }else{

   cout<<endl;
   cout<<"That answer isn't one that was given....oh well..."<<endl;
   cout<<endl;


   name.FName="White ";
   name.MName="The ";
   name.LName="Fox";

 }


name.FulName=name.FName+" "+name.MName+" "+name.LName;

//Output Name to file
ofstream outputFile;
outputFile.open("NameData.txt");


outputFile<<name.FName<<" "<<endl;

outputFile.close();
cout<<endl;
cout<<"Done, name saved!\n";
cout<<endl;
```

```
cout<<"Continue..."<<endl;
cout<<"(Whenever you see this message by itself, type in anything other\n";
cout<<" than a space or tab to move on.)"<<endl;


cout<<"**************************************************************\n";
cout<<"Chapter 0- Intro\n"<<endl;
cout<<"--------------------------------------------------------------\n";

//Output Name in File to terminal
fstream textfile;
textfile.open("NameData.txt");

while(! textfile.eof()){
   textfile>>input[txtName];
   cout<<input[txtName];
   txtName++;
}


cout<<"...You are the great candy making adventurer, "<<name.FulName<<"."<<endl;
cout<<"Amongst your travels, you have heard of a legendary candy bar \n";
cout<<"that, when eaten, will give the consumer the secrets to creating\n";
cout<<"the most amazing confectionary delights the world has ever known.\n";

cout<<endl;
cout<<"Continue..."<<endl;

Continu ();

cout<<"Among your many years of traveling the world in search of amazing\n";
cout<<"sugary artifacts, you have finally gathered enough information to\n";
cout<<"find the location of THE CANDY BAR"<<endl;
cout<<"Continue..."<<endl;

Continu ();


cout<<"**************************************************************\n";
cout<<"Chapter 1- The Forest\n"<<endl;
cout<<"--------------------------------------------------------------\n";

cout<<"You begin to travel out of town and head towards the forest"<<endl;
cout<<"where you heard the candy bar was located. It seems quite\n";
cout<<"for most of the walk."<<endl;

cout<<"Continue..."<<endl;
Continu ();
```

```
cout<<"You're almost at your destination, when suddenly, you hear a loud\n";
cout<<"PLOP behind you."<<endl;

cout<<"Continue..."<<endl;
Continu();

cout<<"A living blob of chocolate and vanilla pudding, easily twice your\n";
cout<<"your size has decided to make YOU its next meal!."<<endl;

cout<<"Continue..."<<endl;
Continu();

//Initiates battle function and checks for a "Game Over"
gameOvr=Battle1 (mcStats[0], mcStats[1], mcStats[3], crit);

//Ends game if main character is defeated
if(gameOvr==1){
   return 0;
}


cout<<"After you deal the final blow on the giant pudding, it wiggles, \n";
cout<<"stays perfectly still for a bit, and then lies flat on the ground\n";
cout<<"as it makes a sound like a deflating whoopee cushion";
cout<<endl;

cout<<"After dealing with the giant pudding, you venture more towards\n";
cout<<"the last known location of the candy bar."<<endl;

cout<<"Continue..."<<endl;
Continu();

cout<<"**************************************************************\n";
cout<<"Chapter 2 - The Cave\n"<<endl;
cout<<"--------------------------------------------------------------\n";

cout<<"After traveling for several hours, you make your way to the cave\n";
cout<<"where the Candy Bar lies. The caves entrance has frosting moss\n";
cout<<"growing around it, a testament to the years the cave has existed\n";

cout<<"Continue..."<<endl;
Continu();

cout<<"Mysteriously, a few lollipop torches catch flame and illuminate\n";
cout<<"the way inside. As an explorer extraordinaire and a seeker of\n";
cout<<"sweets, you confidently stride forth into the cavern"<<endl;
```

```
cout<<"Continue..."<<endl;
Continu();

cout<<"A loud thudding sound comes from behind you. The cave just sealed\n";
cout<<"itself off. A second thud soon follows, this time coming from a\n";
cout<<"large boulder\n";

cout<<"Continue..."<<endl;
Continu();

cout<<"You start running away from the boulder. Its starting to gain on you\n";
cout<<"Luckily there's a ledge ahead of you that you can jump to and get\n";
cout<<"out of the boulders way\n"<<endl;

cout<<"Continue..."<<endl;
Continu();

cout<<"You try to jump and grab the ledge..."<<endl;
cout<<"Continue..."<<endl;
Continu();
//Event with a check if succeeded or not
if(rand()%20+mcStats[3]>17){
   cout<<"You survived!!!"<<endl;
   cout<<"Upon hoisting yourself up, you spot the Candy Bar!"<<endl;
   cout<<"Continue..."<<endl;

   Continu ();
}else{
   cout<<"You got squished by the boulder"<<endl;
   cout<<"Continue..."<<endl;
   Continu ();

   cout<<"Luckily, the boulder seemed to be made out of a ball of powdered\n";
   cout<<"candy, and just disintegrated as soon as it touched you instead"<<endl;
   cout<<"Continue..."<<endl;
   Continu ();
}

//Event that will dictate the ending of the game
cout<<"You finally see it...the fabled Candy Bar...."<<endl;
cout<<"It sits on a pedestal, wrapped in silvery foil"<<endl;
cout<<"Continue..."<<endl;
Continu ();

cout<<"As you approach, you notice the faint outline of what seems to be\n";
cout<<"a magic barrier around your prize."<<endl;
cout<<"Continue..."<<endl;
Continu ();
```

```
    cout<<"Using a bit of your know how about magic, you attempt to focus on\n";
    cout<<"the barrier to deactivate it."<<endl;
    cout<<"Continue..."<<endl;
    Continu ();

    cout<<"As you attempt to do so, all the lollipop torches in the cave go out!\n";
    cout<<"Continue..."<<endl;
    Continu ();

    cout<<"After they spontaneously light up again, a living shadow is staring\n";
    cout<<"you right in the face."<<endl;
    cout<<"Continue..."<<endl;
    Continu ();

    cout<<"It lets out a loud shriek and changes its form to a copy of you."<<endl;
    cout<<"It appears as a humanoid black mass with glowing eyes that slowly"<<endl;
    cout<<"shifts itself around to copy every fine detail of your silhouette"<<endl;
    cout<<"and lunges toward you!"<<endl;
    cout<<"Continue..."<<endl;
    Continu ();


//    bool Battle2 (short, short, short, float,  //A battle with the second enemy
//        short, short, short, float);

    //Initiates battle function and checks for a "Game Over"
    gameOvr=Battle2 (mcStats[0], mcStats[1], mcStats[3], crit,
                cStats[0], cStats[1], cStats[3]);

    //Ends game if main character is defeated
    if(gameOvr==true){
       return 0;
    }

    //Event that will dictate the ending of the game
    if(rand()%20+mcStats[4]>13){
       cout<<"Your experience with magical constructs allows you to"<<endl;
       cout<<"disable the barrier!";
       cout<<"You grab the legendary treat, unwrap it, and take a bite"<<endl;
       cout<<"Continue..."<<endl;
       Continu ();

       cout<<"You immediately are filled with the knowledge to create the best\n";
       cout<<"treats in the world and with this your quest for the Candy Bar succeeds\n";
       cout<<"Continue..."<<endl;
       Continu ();
```

```
    }else{
        cout<<"You fumble your hands slightly and trigger the barriers safety"<<endl;
        cout<<"mechanisms....however..."<<endl;
        cout<<"Continue..."<<endl;
        Continu ();


        cout<<"You try to grab the legendary treat as quick as you can out of instinct!"<<endl;
        cout<<"You unwrap it, and take a bite"<<endl;
        cout<<"Continue..."<<endl;
        Continu ();

        cout<<"It suddenly flies out of your hands at what you would approximate\n";
        cout<<"as "<<LIGHT<<" miles per hour, or basically the speed of light\n";
        cout<<"The speed of it ends up destroying your hand."<<endl;
        cout<<"Continue..."<<endl;
        Continu ();

    }


    cout<<"Years pass by..."<<endl;
    cout<<"Continue..."<<endl;
    Continu ();

    cout<<"You are "<<name.FulName<<" the legendary candy maker."<<endl;
    cout<<"By tasting the legendary Candy Bar, you have created a world devoid\n";
    cout<<"of hunger by creating candies that never spoil and can nourish someone\n";
    cout<<"for a month with just one bite. History remembers you as:"<<endl;
    cout<<endl;
    cout<<name.FulName<<endl;
    cout<<"The legendary candysmith."<<endl;
    cout<<endl;
    cout<<"Continue..."<<endl;
    Continu ();

    cout<<"The End"<<endl;




    //Exit Stage Right
    return 0;
}


//The Title Screen
void TtlScrn (){
```

```
    cout<<"                Fred Roybal IV Presents:                    \n";
    cout<<"***********************************************************\n";
    cout<<"-----------------------------------------------------------\n";
    cout<<"                The    Legend    of                    \n";
    cout<<"                       The                          \n";
    cout<<"              Amazingly    Spectacular                \n";
    cout<<"                     CANDY BAR                       \n";
    cout<<"-----------------------------------------------------------\n";
    cout<<"***********************************************************\n";
    return;
}

//The function used to space out text and wait until the player is ready to continue
void Continu (){

    string Verify;

    cin>>Verify;

    cout<<endl;

}

//Used to calculate if the "Critical Strike" skill in the battle function does
//extra damage and how much
unsigned short CritRt (short base, float rate){

    unsigned short damage;
    unsigned short crit;
    float calc=rand()%99+1.5;   //Randomly generates a number to decide if an
                                //attack critically strikes


    if (calc<=rate){
        //Output notification if succeeded
        cout<<"CRITICAL HIT!!! ";
        damage=base*2;
    }else{
        //Output normal damage if failed
        damage=base;

    }

    return damage;
}

//Function containing the entire code for the battle with the Giant Pudding
//Entered numbers are the main characters statistics
bool Battle1 (short mcHp, short mcStr, short mcMag, float mcCrit){
```

```
//Declare Variables
unsigned char mnstrUp=1;        //Checks if the enemy is still alive

unsigned char Cmd;              //For tracking user input

unsigned char trnOver=0;        //Used for tracking who's turn it is

unsigned char ablMenu;          //Used for tracking which ability a user
                                //might choose

unsigned char ablSlct=0;        //Tracks if the user selected an ability

short pHp=70;                   //Statistics of enemy
unsigned short pStr=8;

bool gameOvr;           //Tracks if the user is defeated


cout<<"The Giant Pudding slaps you!"<<endl;
cout<<"You have "<<(mcHp-=(rand()%19+1)+pStr)<<" health left."<<endl;
cout<<endl;
cout<<"(You are in a battle!)"<<endl;
cout<<"(The battle is over when either you or the enemy is defeated)"<<endl;

cout<<"Continue..."<<endl;
Continu ();

do{

   //Signals for when the player is defeated
   if(mcHp<=0){
      cout<<"You just got defeated by the tutorial monster"<<endl;
      cout<<"Were you even trying?"<<endl;
      cout<<"Continue..."<<endl;
      Continu ();
      cout<<"The Giant Pudding takes your lunch money and skulk"<<endl;
      cout<<"into the sunset. You cease your adventures as a candy making explorer"<<endl;
      cout<<"and go to work in the Chocolate Mines for minimum wage"<<endl;
      cout<<"...for the rest of your life"<<endl;
      cout<<endl;
      cout<<"Continue..."<<endl;
      Continu ();

      cout<<"GAME OVER"<<endl;
      return gameOvr=true;
      cout<<endl;
      ;
   }
```

```
//List of the user's available actions
if(mcHp>0){
cout<<"Menu**********"<<endl;
cout<<"1: Attack"<<endl;
cout<<"2: Use Skill"<<endl;
cout<<"3: Aggressively Breakdance"<<endl;
cout<<"**************"<<endl;
cin>>Cmd;

   switch (Cmd){
      //Basic Attack command
      case '1':
         cout<<"You attack the pudding!"<<endl;
         cout<<"It has "<<endl;
         cout<<(pHp-=(rand()%39+1)+(mcStr))<<endl;
         cout<<"health left"<<endl;
         ablSlct++;
         break;
      //Menu for skill select
      case '2':
         do{
         cout<<"Menu**********"<<endl;
         cout<<"1: Critical Strike"<<endl;
         cout<<"2: Reckless Charge"<<endl;
         cout<<"3: GET PUMPED!!!"<<endl;
         cout<<"**************"<<endl;
         cin>>ablMenu;

         switch (ablMenu){
               case '1':
               cout<<"You focus on hitting a weak point\n";
               cout<<(pHp-=CritRt((rand()%14+1),50))<<endl;
               cout<<"The Giant pudding has "<<pHp<<" health left.";
               ablSlct++;
               break;

               case '2':
               cout<<"You charge head on at the enemy, and over exert";
               cout<<" yourself to deal more damage."<<endl;
               cout<<"The pudding has "<<endl;
               cout<<(pHp-=(rand()%34+20));
               cout<<" health left."<<endl;
               cout<<"You hurt yourself in the attack!"<<endl;
               cout<<"You have "<<(mcHp-=(rand()%4+1))<<" health left"<<endl;
               ablSlct++;
               break;

               case '3':
```

```
            cout<<"You flex your muscles and GET PUMPED!!!"<<endl;
            cout<<"Your strength has increased to ";
            cout<<(mcStr+=(rand()%4+(mcMag-3)))<<endl;
            ablSlct++;
            break;

            //Used for when invalid input is entered from skill menu
            default:
            cout<<"Your head hurts from trying to think"<<endl;
            cout<<"(That isn't a valid response)"<<endl;

            Continu ();
            break;
          }

        }while (ablSlct==0);
        break;
    //A comical action the user can do
    case '3':
        cout<<"You decide to pull off some sweet dance moves"<<endl;
        cout<<"The Giant Pudding just go served"<<endl;
        cout<<rand()%99+1;
        cout<<" style points gained"<<endl;
        ablSlct++;
        break;
    //Used for when invalid input is entered from main menu
    default:
          cout<<"Your head hurts from trying to think"<<endl;
          cout<<"(That isn't a valid response)"<<endl;
          break;

  }
  //Checks if enemy is defeated
  if(pHp<=0){
    --mnstrUp;
    cout<<"You defeated the Giant Pudding!!!"<<endl;
    cout<<"Continue..."<<endl;
    Continu ();
    cout<<"You leveled up to level 44!"<<endl;
    cout<<"Giant Pudding dropped 1043 gold"<<endl;
    cout<<"Continue..."<<endl;
    Continu ();
  }
  //Calculates and tracks damage to user when enemy attacks
   cout<<"The Giant Pudding slaps you!"<<endl;
   cout<<"You have "<<(mcHp-=(rand()%19+1)+pStr)<<" left"<<endl;
   cout<<endl;

}
```

```
    }while(!mnstrUp==0);

    return gameOvr=false;
}

unsigned short Poison (short countP){

    unsigned short counter1;
    unsigned short dmg;

    for (counter1=1;counter1<=countP;counter1++){

        dmg=counter1;
    }

    return dmg;

}

bool Battle2 (short mcHp, short mcStr, short mcMag, float mcCrit,
              short cHp, short cStr, short cMag){

//Declare Variables
    unsigned char mnstrUp=1;        //Checks if the enemy is still alive

    unsigned char Cmd;              //For tracking user input

    unsigned char trnOver=0;        //Used for tracking who's turn it is

    unsigned char ablMenu;          //Used for tracking which ability a user
                                    //might choose

    unsigned char ablSlct=0;        //Tracks if the user selected an ability

    bool gameOvr=false;                 //Tracks if the user is defeated

    unsigned short poison=1;          //Tracks poison damage
    unsigned short count=0;

    cout<<"The Doppleganger is emitting a black smoke from itself!"<<endl;
    cout<<"You have been poisoned!"<<endl;
    cout<<"Continue..."<<endl;
    Continu();

    do{
        for (count=0;count<=poison;count++){
```

```
        mcHp-=count;
    }

    poison++;

    cout<<"The effects of the poison are taking it's toll."<<endl;
    cout<<"You have "<<mcHp<<" health left"<<endl;


    //Signals for when the player is defeated
    if(mcHp<=0){
        cout<<"You were defeated by your doppleganger"<<endl;
        cout<<"Continue..."<<endl;
        Continu ();
        cout<<"The Doppleganger takes your lunch money and your memories"<<endl;
        cout<<"and runs away. You cease your adventures as a candy making explorer"<<endl;
        cout<<"and go to work in the Chocolate Mines for minimum wage"<<endl;
        cout<<"...for the rest of your life"<<endl;
        cout<<endl;
        cout<<"Continue..."<<endl;
        Continu ();

        cout<<"GAME OVER"<<endl;
        return gameOvr=true;
        cout<<endl;
        ;
    }

//List of the user's available actions
if(mcHp>0){
cout<<"Menu**********"<<endl;
cout<<"1: Attack"<<endl;
cout<<"2: Use Skill"<<endl;
cout<<"3: Aggressively Breakdance"<<endl;
cout<<"**************"<<endl;
cin>>Cmd;

    switch (Cmd){
        //Basic Attack command
        case '1':
            cout<<"You attack the Doppleganger!"<<endl;
            cout<<"It has "<<endl;
            cout<<(cHp-=(rand()%39+1)+(mcStr))<<endl;
            cout<<"health left"<<endl;
            ablSlct++;
            break;
        //Menu for skill select
        case '2':
            do{
```

```
    cout<<"Menu**********"<<endl;
    cout<<"1: Critical Strike"<<endl;
    cout<<"2: Reckless Charge"<<endl;
    cout<<"3: GET PUMPED!!!"<<endl;
    cout<<"**************"<<endl;
    cin>>ablMenu;

    switch (ablMenu){
          case '1':
          cout<<"You focus on hitting a weak point\n";
          cout<<(cHp-=CritRt((rand()%14+1),50))<<endl;
          cout<<"The Giant pudding has "<<cHp<<" health left.";
          ablSlct++;
          break;

          case '2':
          cout<<"You charge head on at the enemy, and over exert";
          cout<<" yourself to deal more damage."<<endl;
          cout<<"The pudding has "<<endl;
          cout<<(cHp-=(rand()%34+20));
          cout<<" health left."<<endl;
          cout<<"You hurt yourself in the attack!"<<endl;
          cout<<"You have "<<(mcHp-=(rand()%4+1))<<" health left"<<endl;
          ablSlct++;
          break;

          case '3':
          cout<<"You flex your muscles and GET PUMPED!!!"<<endl;
          cout<<"Your strength has increased to ";
          cout<<(mcStr+=(rand()%4+(mcMag-3)))<<endl;
          ablSlct++;
          break;

          //Used for when invalid input is entered from skill menu
          default:
          cout<<"Your head hurts from trying to think"<<endl;
          cout<<"(That isn't a valid response)"<<endl;

          Continu ();
          break;
       }

    }while (ablSlct==0);
    break;
//A comical action the user can do
case '3':
    cout<<"You decide to pull off some sweet dance moves"<<endl;
    cout<<"The Doppleganger just go served"<<endl;
    cout<<rand()%99+1;
```

```
            cout<<" style points gained"<<endl;
            ablSlct++;
            break;
         //Used for when invalid input is entered from main menu
         default:
               cout<<"Your head hurts from trying to think"<<endl;
               cout<<"(That isn't a valid response)"<<endl;
               break;

      }
      //Checks if enemy is defeated
      if(cHp<=0){
         --mnstrUp;
         cout<<"You defeated the Doppleganger!!!"<<endl;
         cout<<"Continue..."<<endl;
         Continu ();
         cout<<"You leveled up to level 103!"<<endl;
         cout<<"The Doppleganger dropped 3432 gold and some black licorice"<<endl;
         cout<<"Continue..."<<endl;
         Continu ();
      }
      //Calculates and tracks damage to user when enemy attacks
       cout<<"The Doppleganger strikes you!"<<endl;
       cout<<"You have "<<(mcHp-=(rand()%19+1)+cStr)<<" left"<<endl;
       cout<<endl;

    }


   }while(!mnstrUp==0);

   return gameOvr=false;
}

short Stub (short a){

   cout<<"Stub test successful"<<endl;
   cout<<"calling function Poison"<<endl;
   cout<<Poison (a)<<endl;
   cout<<"Poison test end"<<endl;

}
```