Project 2

<RPG Game: Dragon Hero v2>

CSC-5 40717
Name: Justin Waterman
Date: 02/12/2015

Introduction

Title: Dragon Hero

This is a small RPG (role-playing game).
RPG games include the individual creation of a character customized to the user's choice.
The user can then play as his or her own custom character in a world where he or she makes decisions in the character's world. Those decisions can then affect how the rest of the game plays out and ends.

This game takes place in a kingdom from long ago when dragons once lived. The main character, the user, is a well-known dragon slayer and must seek out a dragon that has been causing havoc in the kingdom. The main character must journey from his home, in search of this dragon and must survive the perils that this adventure entails. When the dragon is slain, the kingdom is saved.

The game revolves around 'lifepoints' and rolling dice.
Lifepoints are, essentially, the main character's life. When the main character loses all of his or her lifepoints, the player dies. Thus, the game is over and the user loses.
In order to determine how to the game plays out, rolling dice are used.
The user will 'roll dice' and the number rolled will determine how and where the main character's adventures plays.

Summary

Project size: Around 750 lines.
The number of variables used: about 50

I tried to create a role-playing game, where the user's choices and decisions can win or lose the game, while also adding a small amount of randomness to help the game be more challenging. Being a role-playing game, this program has endless potential for concepts, stories, and functions to be added on in the future. While the game may feel short when played, I plan to work on it further to hopefully have a much more immersive experience.

C++ can be simple at times, but can quickly become overwhelmingly complicated if not coded and understood correctly.

The most time consuming errors of my project concerned strings and the cin input. For a few days, my program would execute entirely without allowing me to enter values. The program would simply skip the cin statements. I later came some sort of a resolution by converting most of my strings to more primitive data types, such as ints or shorts.

Another time consuming piece of my game was the amount of revisions and changed made to it. I would begin to code something up, only to realize I had gone over my head, unable to perform my desired actions with my level of experience with C++.

I finally stopped with the current revision due to it not being overly complicated, but still having enough if statements and loops to give a relatively new coder a headache to debug.

Rev 2
My second major revision to this game for my project included adding multiple functions, arrays, and sorting.

The original version of this game was all contained into the main function. I separated out many parts of the program into their own functions, which helps to declutter the main program as well as to make troubleshooting much easier. I also added another small mini-game within the program which involved the sorting and switching of integer arrays. I utilized the Mark Sort method to sort my integer array, which is made famous by Dr. Mark Lehr.

The program will not show much change when run, however, because most of the changes are done under the hood.

Description
The main point of this program is that if statements and loops are very powerful tools and are capable of running a decision-heavy game.

Pseudo Code
Initialize
Prompt user for character creation in the charcre function
Write character information to text file stats.txt
Roll dice to determine journey
If dice is 5 or 6,
Run roll65 function
    User goes through jungle and fights tiger.
        If user loses, decrease lifepoint
    User then approaches a bridge and must solve a math problem
        If user loses, decrease lifepoint
Else if dice is 3 or 4,
Run roll43 function
    User approaches a bridge and must solve a math problem
        If user loses, decrease lifepoint
Else if dice is 1 or 2
    User doesn't need to travel anywhere
User approaches dragon's lair
Roll dice
If dice is 5 or 6,
Run dg65 function
    User can kill sleeping dragon or crack a code to banish the dragon from the kingdom
        If user loses, decrease 3 lifepoints
If dice is 3 or 4,
Run dg43 function

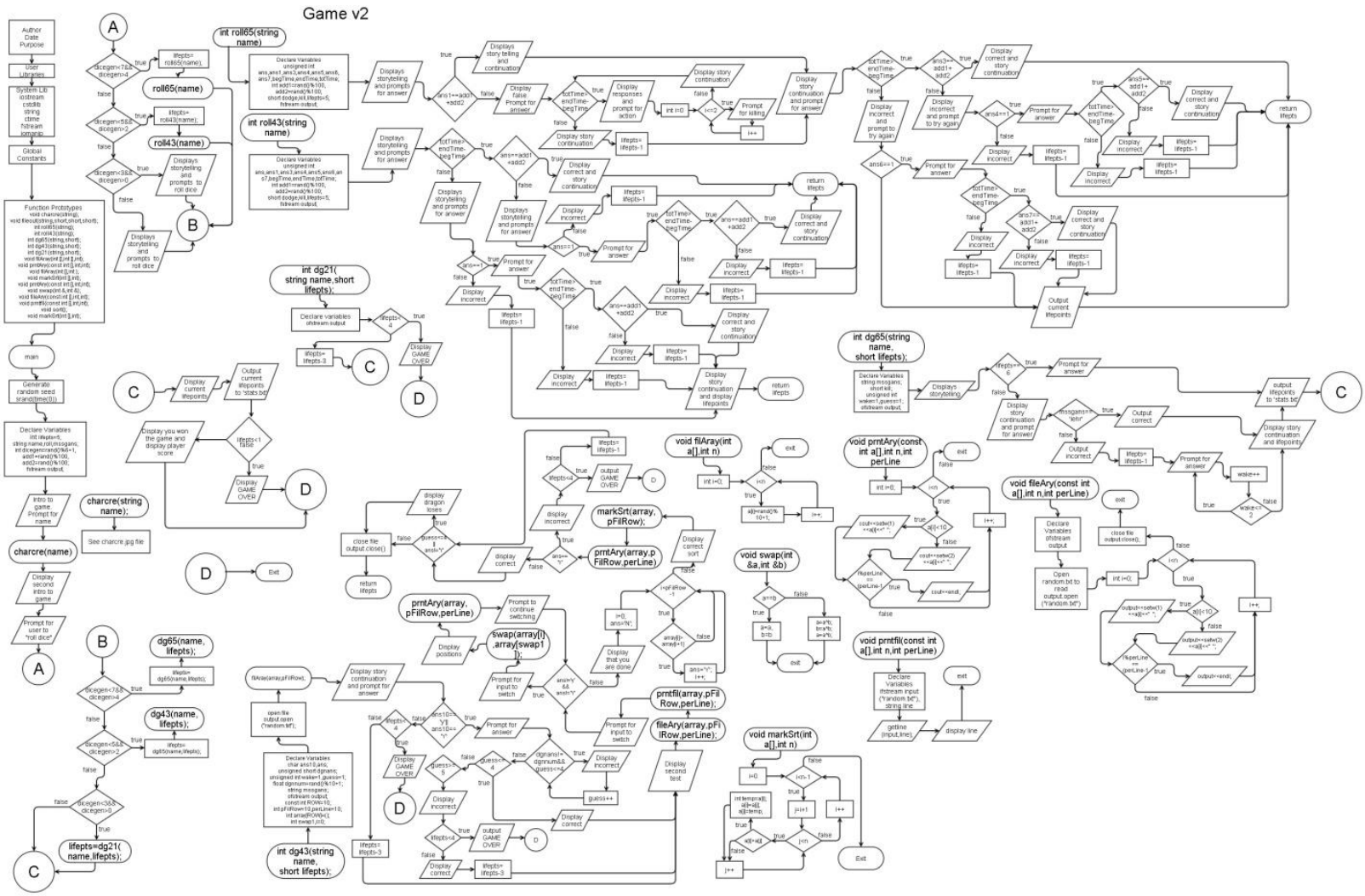
User finds dragon and must answer its question to banish the dragon or fight it
If user loses, decrease 3 lifepoints
User plays dragon's second test.
If user loses, decrease 3 lifepoints
If dice is 1 or 2
User must fight dragon until the dragon or user is dead.
Display player's score
If lifepoints<1 then user loses the game
Shown game ending

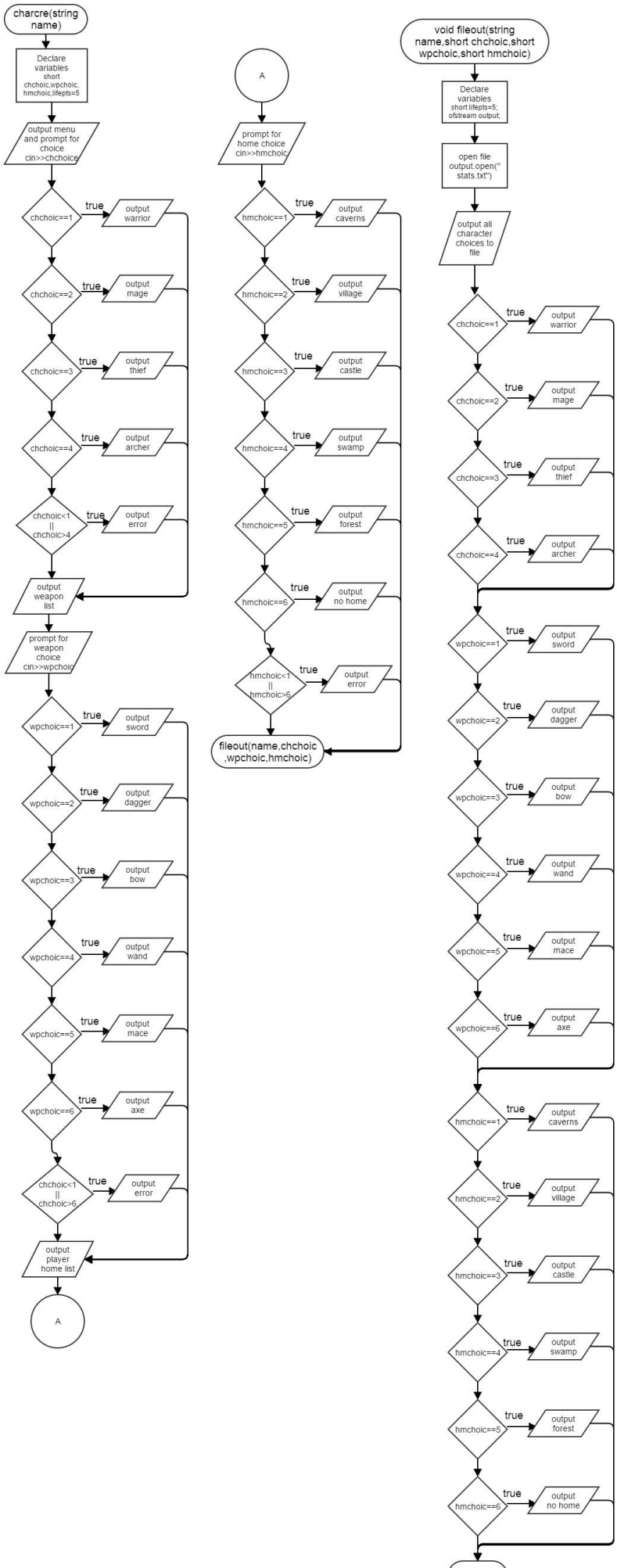
Variables

Data type	Name	Description
Char	ans10	Answer to a yes or no question
Int	Add1	Random number for math question
	Add2	Random number for math question
	Dicegen	Random number for rolling dice
	pFilRow	Length of array
	perLine	Length of array
	Array[]	Array
	Swap1	Number to swap with
Unsigned Int	Ans1	Answer to math question
	Ans3	Answer to math question
	Ans4	Answer to yes or no question
	Ans5	Answer to math question
	Ans6	Answer to yes or no question
	Ans7	Answer to math question
	begTime	Starting variable for time based answers
	endTime	Ending variable for time based answers
	totTime	Total amount of time allowed to answer certain questions
	wake	Used as a counter in a loop
	guess	Used as counter for number of guesses
String	Name	Character's name
	Roll	Variable used to begin a dice roll
	Mssgans	Variable for encrypted message
	Line	Used to display a line from a file
Unsigned short	dgnans	Answer to dragon's question
float	dgnnum	Random number dragon is thinking of
short	chchoic	Character type choice
	wpchoic	Character weapon choice
	hmchoic	Character home choice

	dodge	Time based answer to dodge attack
	kill	Variable to kill enemy
	lifepts	Holds number of lifepoints the main character has
fstream	output	Output stats to a text file
Const int	ROW	Length of array
Ifstream	Input	Used to read a file

Flowchart




charcre(string name)
Declare variables short chchoic,wpchoic, hmchoic,lifepts=5
output menu and prompt for choice cin>>chchoice
chchoic==1
true
output warrior
chchoic==2
true
output mage
chchoic==3
true
output thief
chchoic==4
true
output archer
chchoic<1 || chchoic>4
true
output error
output weapon list
prompt for weapon choice cin>>wpchoic
wpchoic==1
true
output sword
wpchoic==2
true
output dagger
wpchoic==3
true
output bow
wpchoic==4
true
output wand
wpchoic==5
true
output mace
wpchoic==6
true
output axe
chchoic<1 || chchoic>6
true
output error
output player home list
A
A
prompt for home choice cin>>hmchoic
hmchoic==1
true
output caverns
hmchoic==2
true
output village
hmchoic==3
true
output castle
hmchoic==4
true
output swamp
hmchoic==5
true
output forest
hmchoic==6
true
output no home
hmchoic<1 || hmchoic>6
true
output error
fileout(name,chchoic ,wpchoic,hmchoic)
void fileout(string name,short chchoic,short wpchoic,short hmchoic)
Declare variables short lifepts=5; ofstream output;
open file output.open(" stats.txt")
output all character choices to file
chchoic==1
true
output warrior
chchoic==2
true
output mage
chchoic==3
true
output thief
chchoic==4
true
output archer
wpchoic==1
true
output sword
wpchoic==2
true
output dagger
wpchoic==3
true
output bow
wpchoic==4
true
output wand
wpchoic==5
true
output mace
wpchoic==6
true
output axe
hmchoic==1
true
output caverns
hmchoic==2
true
output village
hmchoic==3
true
output castle
hmchoic==4
true
output swamp
hmchoic==5
true
output forest
hmchoic==6
true
output no home

C++ Constructs

Chapter	Name	A Location
2	Cout Object	Cout<<"Welcome to my game!"<<endl;
	#include Directive	#include <iostream>
	Variables	Char ans10;
	Integer Data Types	Int add1=rand()%100;
	Char Data Types	Char ans10;
	String Class	String name;
	Floating point Data Types	Float dgnnum=rand()%10+1
	Scope	String name;
	Arithmetic Operators	If(ans3==add1+add2)
	Comments	//Message means 'lehr'
3	Cin Object	Cin>>ans;
	Mathematical Expressions	If(ans3==add1+add2)
	Type casting	begTime=static_cast<unsigned int>(time(0));
	Formatting Output	cout<<"Welcome "<<name<<". "<<"What would you like to be?\n1."<<setw(2)<<"A Warrior\n2. "<<endl;
4	Relational Operators	If(totTime>endTime-begTime)
	If Statements	If(dicegen<7&&dicegen>4)
	The if/else Statement	Else if(dicegen>5&&dicegen>2

	Nested if Statements	If(totTime>endTime-begTime){
		If(ans7==add1+add2)
		}

	Menus	cout<<"Weapon list:"<<endl; cout<<"1. Sword"<<endl; cout<<"2. Dagger"<<endl;
	Switch statement	Switch(wpchoic){
5	Increment and Decrement Operators	For(int i=o;i<=2;i++)
	While Loop	While(dgnans!=dgnnum&&guess<=4)
	Counters	Guess++;

	Do-While Loop	do{
		cout<<"Attackit with your weapon! Enter 1!"<<endl;
		cin>>kill;
		wake++;
		}while(wake<=2);

	For Loop	for(int i=0;i<=2;i++){ cin>>kill; cout<<"Keep attacking!

		Press 2"<<endl; }
6	Defining and Calling Functions	fileout(name,chchoic,wpchoic,hmchoic);
	Function Prototypes	void charcre(string);
	Sending Data into Function	int roll65(string);
	Passing Data by Value	dg65(string name,short lifepts)
	Return value from function	return lifepts;
	Default Arguments	void prntAry(const int [],int=10,int=10);
	Using reference variables as arguments	void swap(int &,int &);
7	Arrays	int array[ROW]={};
	Arrays as Function Arguments	void filAray(int a[],int n)
	Sorting	void markSrt(int a[],int n); void swap(int &a,int &b);
Other	Writing to File	ofstream output; output.open("stats.txt"); output<<"Your name is "<<name
	Reading a File	ifstream input ("random.txt"); string line; getline(input,line);

Reference

1. Textbook

Program

```
#include <cstdlib>
#include <iostream>
#include <string>
#include <ctime>
#include <fstream>
#include <iomanip>
using namespace std;
//User Libraries
//Global Constants
//Function Prototypes
void charcre(string); //Character Creation
void fileout(string,short,short,short); //Send character stats out to file
int roll65(string);  //If roll is 6 or 5
int roll43(string);  //If roll is 3 or 4
int dg65(string,short); //If roll is 6 or 5
int dg43(string,short); //If roll is 3 or 4
int dg21(string,short); //If roll is 1 or 2
void prntAry(const int [],int,int);  //Print array
void filAray(int [],int ); //Fill array
void markSrt(int  [],int); //Sorting
void swap(int &,int &); //Swap two numbers
```

```
void fileAry(const int [],int,int); //Read file
void prntfil(const int [],int,int); //Print file
//Execution
int main(){
   //Generate random seed
   srand(static_cast<unsigned int>(time(0)));
   //Declare Variables
   short lifepts=5;
   string name,roll,mssgans;
   int dicegen=rand()%6+1, add1=rand()%100, add2=rand()%100;
   fstream output;
   cout<<"Welcome to my game!"<<endl;
   cout<<"What is your character's name? (Just first name, please)"<<endl;
   cin>>name;
   //Create character
   charcre(name);
   cout<<"Here is where we begin our adventure.\n"<<endl;
   cout<<"----------------------------------------\n"<<endl;
   cout<<"You live in a kingdom called "<<name<<"-land."<<endl;
   cout<<"Here, you are a great dragon killer."<<endl;
   cout<<"A new dragon has made its home somewhere in this area.";
   cout<<" You must find and kill this dragon."<<endl;
   cout<<"In order to begin, pull out the magic dice that are in your pocket and roll";
   cout<<" them to see where your journey will begin.\nYou also have 5 Lifepoints. If damaged in combat"
         " or attack or wounded, you will lose a lifepoint.\nIf you reach 0 Lifepoints, then"
         " GAME OVER\n"<<endl;
   cout<<"Input 'roll' to roll your dice."<<endl;
   //Roll Dice
   cin>>roll;
   cout<<"You roll a "<<dicegen<<endl;
   //If dice roll is 5 or 6
   if(dicegen<7&&dicegen>4)lifepts=roll65(name);
   //If dice roll is 3 or 4
   if(dicegen<5&&dicegen>2)lifepts=roll43(name);
   //If dice roll is 1 or 2
   if(dicegen<3&&dicegen>0)
      cout<<"This means you are close enough to the dragon's lair that a journey is not needed."<<endl;
   cout<<"You finally reach the dragon's lair.\nNow you enter the dragon's lair."<<endl;
   cout<<"Roll your dice to determine how you find the dragon."<<endl;
   //Roll dice
   cin>>roll;
   cout<<dicegen<<endl;
   //If dice roll is 5 or 6
   if(dicegen<7&&dicegen>4)lifepts=dg65(name,lifepts);
   //If dice roll is 3 or 4
   if(dicegen<5&&dicegen>2)lifepts=dg43(name,lifepts);
   //If dice roll is 1 or 2
   if(dicegen<3&&dicegen>0)lifepts=dg21(name,lifepts);
   cout<<"[Your current lifepoints are "<<lifepts<<" ]"<<endl;
   //Output lifepoints to text file
   output<<"Your current lifepoints are "<<lifepts<<". \r\n"<<endl;
   cout<<endl;
   //If user's final lifepoints are zero
   if(lifepts<1){
      cout<<"You have no lifepoints remaining. GAME OVER"<<endl;
      return 0;
```

```
    }
    //If user has more than zero lifepoints
    else{
        cout<<"Congratulations "<<name<<"! You killed the dragon, saved the world, and won my game."
            " Thank you for playing\nYour score is "<<lifepts<<". (This is based off of your lifepoints.)"<<endl;
    }
    //Close file
    output.close();
    //Exit
    return 0;
}
void charcre(string name){
    short chchoic,wpchoic,hmchoic,lifepts=5;
    //Character Creation
    cout<<"Let's begin by creating your character."<<endl;
    cout<<"Welcome "<<name<<". "<<"What would you like to be?\n1."<<setw(2)<<"A
Warrior\n2."<<setw(2)<<"A Mage\n3."<<setw(2)<<"A Thief\n"
          "4."<<setw(2)<<"An Archer\n"<<endl;
    cout<<"Input the number of your preferred character."<<endl;
    //Choice your type of character
    cin>>chchoic;
    cout<<endl;
    if(chchoic==1){
        cout<<"Congratulations you are a Warrior!"<<endl;
        cout<<"\nSelect your weapon!"<<endl;
    }
    else if(chchoic==2){
        cout<<"Congratulations you are a Mage!"<<endl;
        cout<<"\nSelect your weapon!"<<endl;
    }
    else if(chchoic==3){
        cout<<"Congratulations you are a Thief!"<<endl;
        cout<<"\nSelect your weapon!"<<endl;
    }
    else if(chchoic==4){
        cout<<"Congratulations you are an Archer!"<<endl;
        cout<<"\nSelect your weapon!"<<endl;
    }
    else if(chchoic<1 || chchoic>4){
        cout<<"Error. Please enter a number corresponding to the above menu."<<endl;
    }
    cout<<"Weapon list:"<<endl;
    cout<<"1. Sword"<<endl;
    cout<<"2. Dagger"<<endl;
    cout<<"3. Bow"<<endl;
    cout<<"4. Wand"<<endl;
    cout<<"5. Mace"<<endl;
    cout<<"6. Axe"<<endl;
    cout<<"Select your weapon!"<<endl;
    //Choose your weapon
    cin>>wpchoic;
    cout<<endl;
    switch(wpchoic){
        case 1:{
            cout<<"You now wield a sword!"<<endl;
            break;
```

```
    }
    case 2:{
       cout<<"You now wield a dagger!"<<endl;
       break;
    }
    case 3:{
       cout<<"You now wield a Bow!"<<endl;
       break;
    }
    case 4:{
       cout<<"You now wield a Wand!"<<endl;
       break;
    }
    case 5:{
       cout<<"You now wield a Mace!"<<endl;
       break;
    }
    case 6:{
       cout<<"You now wield an Axe!"<<endl;
       break;
    }
    default:{
       cout<<"Please select a weapon in the list."<<endl;
       break;
    }
}
cout<<endl;
cout<<"Where do you live?"<<endl;
cout<<"1. Caverns\n2. Small Village\n3. Castle\n4. Swamp\n5. Forest\n"
      "6. You have no home. You wander aimlessly around the world."<<endl;
//Choose your home
cin>>hmchoic;
cout<<endl;
switch(hmchoic){
    case 1:{
       cout<<"You now live in dark caverns!"<<endl;
       break;
    }
    case 2:{
       cout<<"You now live in a small village!"<<endl;
       break;
    }
    case 3:{
       cout<<"You now live in a large Castle!"<<endl;
       break;
    }
    case 4:{
       cout<<"You now live in a humid swamp!"<<endl;
       break;
    }
    case 5:{
       cout<<"You now wield a green forest!"<<endl;
       break;
    }
    case 6:{
       cout<<"You now have no home and wander from land to land!"<<endl;
```

```
            break;
        }
        default:{
            cout<<"Please select a home in the list."<<endl;
            break;
        }
    }
    cout<<"\n"<<endl;
    //Write character info to a file
    fileout(name,chchoic,wpchoic,hmchoic);
    //Exit
}
void fileout(string name,short chchoic,short wpchoic,short hmchoic){
    short lifepts=5;
    ofstream output;
    cout<<"[All of your character information as well as your lifepoints will be recorded to a text file called
'stats'.]"<<endl;
    //Output to stats file
    output.open("stats.txt");
    //Output the corresponding character trait to stats file
    output<<"Your name is "<<name<<". \r\n"<<endl<<endl;
    if(chchoic==1)
        output<<"You are a warrior. \r\n"<<endl<<endl;
    else if(chchoic==2)
        output<<"You are a mage. \r\n"<<endl<<endl;
    else if(chchoic==3)
        output<<"You are a thief. \r\n"<<endl<<endl;
    else if(chchoic==4)
        output<<"You are an Archer. \r\n"<<endl<<endl;
    if(wpchoic==1)
        output<<"You wield a sword. \r\n"<<endl<<endl;
    else if(wpchoic==2)
        output<<"You wield a dagger. \r\n"<<endl<<endl;
    else if(wpchoic==3)
        output<<"You wield a bow. \r\n"<<endl<<endl;
    else if(wpchoic==4)
        output<<"You wield a wand. \r\n"<<endl<<endl;
    else if(wpchoic==5)
        output<<"You wield a mace. \r\n"<<endl<<endl;
    else if(wpchoic==6)
        output<<"You wield an axe. \r\n"<<endl<<endl;
    if(hmchoic==1)
        output<<"You live in dark caverns. \r\n"<<endl<<endl;
    else if(hmchoic==2)
        output<<"You live in a small village. \r\n"<<endl<<endl;
    else if(hmchoic==3)
        output<<"You live in a large castle. \r\n"<<endl<<endl;
    else if(hmchoic==4)
        output<<"You live in a humid swamp. \r\n"<<endl<<endl;
    else if(hmchoic==5)
        output<<"You live in a green forest. \r\n"<<endl<<endl;
    else if(hmchoic==6)
        output<<"You now have no home and wander from land to land. \r\n"<<endl<<endl;
    //Exit
}
int roll65(string  name){
```

```
unsigned int ans1,ans3,ans4,ans5,ans6,ans7,begTime,endTime,totTime;
int add1=rand()%100,  add2=rand()%100;
short dodge,kill,lifepts=5;
fstream output;
cout<<"This means you have a long journey ahead of you."<<endl;
   cout<<"You begin traveling through a dark forest. On your travels, you come across a tiger. It looks hungry.\n"
         "You take out your weapon and ready yourself."<<endl;
   cout<<"The tiger then begins to speak!\nIt asks 'What is "<<add1<<" + "<<add2<<" ?"'<<endl;
   //Answer
   cin>>ans1;
   //If answer is correct
   if(ans1==add1+add2)
      cout<<"'You may pass through.'"<<endl;
   //If answer is incorrect
   else{
      cout<<"'Incorrect. Prepare to die!'"<<endl;
      //Timed Response
      begTime=static_cast<unsigned int>(time(0));
      cout<<"The tiger lunges at you.\nTo dodge, press 1. Quickly!"<<endl;
      cin>>dodge;
      endTime=static_cast<unsigned int>(time(0));
      //Total time allowed for user to input something
      totTime=15;
      //If user was fast enough
      if(totTime>endTime-begTime){
         begTime=(time(0));
         cout<<"You dodged the tiger! Quickly, kill it with your weapon!\nInput 2."<<endl;
         //Loop attacking the tiger
         for(int i=0;i<=2;i++){
            cin>>kill;
            cout<<"Keep attacking! Press 2"<<endl;
         }
         cout<<"The tiger is dead! You can now move along on your journey."<<endl;
      }
      //If user was not fast enough
      else{
         cout<<"Too slow! The tiger bites you and takes away one of you life points."<<endl;
         cout<<"You manage you kill it while it is biting you. And you carry on with your"
               "journey, but with one less life point."<<endl;
         lifepts=lifepts-1;
         cout<<"You currently have "<<lifepts<<" lifepoints remaining."<<endl;
      }
   }
   cout<<"\nYou safely leave the forest now and must cross a bridge to continue on your journey."<<endl;
   cout<<"There stands a tall guardian blocking your way. As you approach he asks you a question"
         " in order to cross the bridge. He says you only have 20 seconds to answer the question."<<endl;
   cout<<"Are you ready?"<<endl;
   //Timed based response
   begTime=(time(0));
   cout<<"Hello "<<name<<".\n What is "<<add1<<" + "<<add2<<" ?"<<endl;
   cin>>ans3;
   cout<<ans3<<endl;
   endTime=(time(0));
   //Total time allowed for user to input something
   totTime=20;
   //If user was fast enough
```

```
if(totTime>endTime-begTime){
   //If user was correct
   if(ans3==add1+add2)
      cout<<"'Correct. You may now pass the bridge.'"<<endl;
   //If user was incorrect
   else{
      cout<<"'Incorrect. You may NOT cross the bridge. You may have one more chance to answer a question "
            "correctly or must seek another way around the bridge, which will cost you a lifepoint for the "
            "extra amount of time it will take.'\nInput 1 to try again or 2 to give up a lifepoint and"
            " find another way around."<<endl;
      cin>>ans4;
      //If user wants to try again
      if(ans4==1){
         //Timed response
         begTime=(time(0));
         cout<<"'Your second and final chance.\nWhat is "<<add1<<" + "<<add2<<" ?"<<endl;
         cin>>ans5;
         endTime=(time(0));
         //If user was fast enough
         if(totTime>endTime-begTime){
            //If user was correct
            if(ans5==add1+add2)
               cout<<"'Correct. You may pass the bridge.'"<<endl;
            //If user was incorrect
            else{
               cout<<"'Incorrect. You must find another way around the bridge and you lose 1 lifepoint.'"<<endl;
               //Take away one lifepoint
               lifepts=lifepts-1;
               cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
            }
         }
         //If user was too slow
         else{
            cout<<"'Incorrect. You must find another way around the bridge and you lose 1 lifepoint.'"<<endl;
            //Take away one lifepoint
            lifepts=lifepts-1;
            cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
         }
      }
      //If user didn't want to try again
      else{
         cout<<"'You must find another way around the bridge and you lose 1 lifepoint.'"<<endl;
         //Take away one lifepoint
         lifepts=lifepts-1;
         cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
      }
     }
  }
//If user was not fast enough
else{
   cout<<"You took too long to answer my questions."<<endl;
   cout<<"'Incorrect. You may NOT cross the bridge. You may have one more chance to answer a question "
         "correctly or must seek another way around the bridge, which will cost you a lifepoint for the "
         "extra amount of time it will take.'\nInput 1 to try again or 2 to give up a lifepoint and"
         "find another way around."<<endl;
   cin>>ans6;
```

```
        //If user wants to play again
        if(ans6==1){
          //Time response
          begTime=(time(0));
          cout<<"'Your second and final chance.\nWhat is "<<add1<<" + "<<add2<<" ?"<<endl;
          cin>>ans7;
          endTime=(time(0));
          //If user was fast enough
          if(totTime>endTime-begTime){
            //If user was correct
            if(ans7==add1+add2)
              cout<<"'Correct. You may pass the bridge.'"<<endl;
            //If user was incorrect
            else{
              cout<<"'Incorrect. You must find another way around the bridge and you lose 1 lifepoint.'"<<endl;
              //Take away one lifepoint
              lifepts=lifepts-1;
              cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
            }
          }
          //If user was too slow
          else{
            cout<<"'You still took too long to answer. You must lose a lifepoint and find another way around this
bridge.'"<<endl;
            //Take away one lifepoint
            lifepts=lifepts-1;
            cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
          }
        }
        //If user doesn't want to try again
        else{
          cout<<"'Incorrect. You must find another way around the bridge and you lose 1 lifepoint.'"<<endl;
            //Take away one lifepoint
            lifepts=lifepts-1;
            cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
        }
      }
      //Output lifepoints to text file
      output<<"Your current lifepoints are "<<lifepts<<". \r\n"<<endl;
      //Exit
      return lifepts;
    }
int roll43(string name){
    unsigned int ans,begTime,endTime,totTime;
    int add1=rand()%100, add2=rand()%100;
    short lifepts=5;
    fstream output;
    cout<<"This means you have a 1 day hike to travel."<<endl;
    cout<<"You see a forest to your left, but manage to traverse it with ease. Then you begin to approach "
          "a large bridge that blocks your path between yourself and the dragon's lair."<<endl;
    cout<<"There stands a tall guardian blocking your way. As you approach he asks you a question"
            " in order to cross the bridge. He says you only have 20 seconds to answer the question."<<endl;
    //Timed response
    begTime=(time(0));
    cout<<"Hello "<<name<<".\n What is "<<add1<<" + "<<add2<<" ?"<<endl;
    cin>>ans;
```

```
    endTime=(time(0));
    totTime=20;
    //If user was fast enough
    if(totTime>endTime-begTime){
       //If user was correct
       if(ans==add1+add2)
          cout<<"'Correct. You may pass the bridge.'"<<endl;
       //If user was incorrect
       else{
          cout<<"'Incorrect. You may NOT cross the bridge. You may have one more chance to answer a question "
                "correctly or must seek another way around the bridge, which will cost you a lifepoint for the "
                "extra amount of time it will take.'\nInput 1 to try again or 2 to give up a lifepoint and"
                "find another way around."<<endl;
          cin>>ans;
          //If user wants to try again
          if(ans==1){
             //Timed response
             begTime=(time(0));
             cout<<"'Your second and final chance.\nWhat is "<<add1<<" + "<<add2<<" ?"<<endl;
             cin>>ans;
             endTime=(time(0));
             //If user was fast enough
             if(totTime>endTime-begTime){
                //If user was correct
                if(ans==add1+add2)
                   cout<<"'Correct. You may pass the bridge.'"<<endl;
                //If user was incorrect
                else{
                   cout<<"'Incorrect. You must find another way around the bridge and you lose 1 lifepoint.'"<<endl;
                   //Take away one lifepoint
                   lifepts=lifepts-1;
                   cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
                }
             }
             //If user was too slow
             else{
                cout<<"'You took too long. You must find another way around the bridge and you lose 1
lifepoint.'"<<endl;
                //Take away one lifepoint
                lifepts=lifepts-1;
                cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
             }
          }
          //If user didn't want to try again
          else{
             cout<<"'You must find another way around the bridge and you lose 1 lifepoint.'"<<endl;
                //Take away one lifepoint
                lifepts=lifepts-1;
                cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
          }
       }
    }
    //If user was too slow
    else{
       cout<<"You took too long to answer my questions."<<endl;
       cout<<"'Incorrect. You may NOT cross the bridge. You may have one more chance to answer a question "
```

```
            "correctly or must seek another way around the bridge, which will cost you a lifepoint for the "
            "extra amount of time it will take.'\nInput 1 to try again or 2 to give up a lifepoint and"
            " find another way around."<<endl;
        cin>>ans;
        //If user wants to try again
        if(ans==1){
          //Timed response
          begTime=(time(0));
          cout<<"'Your second and final chance.\nWhat is "<<add1<<" + "<<add2<<" ?"<<endl;
          cin>>ans;
          endTime=(time(0));
          //If user was fast enough
          if(totTime>endTime-begTime){
            //If user was correct
            if(ans==add1+add2)
              cout<<"'Correct. You may pass the bridge.'"<<endl;
            //If user was incorrect
            else{
              cout<<"'Incorrect. You must find another way around the bridge and you lose 1 lifepoint.'"<<endl;
              //Take away one lifepoint
              lifepts=lifepts-1;
              cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
            }
          }
          //If user was too slow
          else{
            cout<<"'You must find another way around the bridge and you lose 1 lifepoint.'"<<endl;
            //Take away one lifepoint
            lifepts=lifepts-1;
            cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
          }
        }
        //If user did not want to try again
        else{
          cout<<"'You must find another way around the bridge and you lose 1 lifepoint.'"<<endl;
          //Take away one lifepoint
          lifepts=lifepts-1;
          cout<<"[Current lifepoints "<<lifepts<<" ]"<<endl;
        }
    }
    cout<<"Once on the other side of the bridge, the dragon's lair is finally in sight."<<endl;
    //Output lifepoints to text file
    output<<"Your current lifepoints are "<<lifepts<<". \r\n"<<endl;
    //Exit
    return lifepts;
}
int dg65(string name,short lifepts){
    string mssgans;
    short kill;
    unsigned int wake=1,guess=1;
    ofstream output;
    cout<<"You find the dragon asleep. If you have all of your lifepoints, you should be able to silently slay the
dragon."<<endl;
    //If user has full lifepoints, he can slay dragon
    if(lifepts==5){
      cout<<"You do, in fact, have full lifepoints. Enter kill to slay the dragon!"<<endl;
```

```
        cin>>kill;
        cout<<"It wakes up in screaming pain as it slowly dies until the dragon is no more."<<endl;
    }
    //If user has less than 5 lifepoints
    else{
        cout<<"You don't have a full amount of lifepoints.\nHowever, you notice strange words on the wall behind the
dragon."<<endl;
        cout<<"It says:\n\t17,10,13,23 \nWhen decoded, this chant will kill the dragon."
            "\nThere is only one clue to decode this message: A=6"<<endl;
        //Message means "lehr"
        cout<<"Enter the message now."<<endl;
        cin>>mssgans;
        //Verify string
        cout<<"You said "<<mssgans<<endl;
        if(mssgans=="lehr")
           cout<<"You cracked the code! Now that you say it, the dragon appears to die until it no longer
moves."<<endl;
        else if(mssgans=="LEHR")
           cout<<"You cracked the code! Now that you say it, the dragon appears to die until it no longer
moves."<<endl;
        else{
           cout<< "Error: wrong code. The dragon wakes up from your talking and begins to attack you!\n"
                "you lose a lifepoint while fighting him."<<endl;
           //Take away a lifepoint
           lifepts=lifepts-1;
           //Loop attack dragon
           do{
              cout<<"Attack it with your weapon! Enter 1!"<<endl;
              cin>>kill;
              wake++;
           }while(wake<=2);
        }
        cout<<"The dragon is dead!"<<endl;
        cout<<"[Your current lifepoints are "<<lifepts<<" ]"<<endl;
        //Output lifepoints to text file
        output<<"Your current lifepoints are "<<lifepts<<". \r\n"<<endl;
    }
    output<<"Your current lifepoints are "<<lifepts<<". \r\n"<<endl;
    //Exit
    return lifepts;
}
int dg43(string name,short lifepts){
    char ans10,ans;
    unsigned short dgnans;
    unsigned int wake=1,guess=1;
    float dgnnum=rand()%10+1;
    string mssgans;
    ofstream output;
    const int ROW=10;
    int pFilRow=10,perLine=10;
    int array[ROW]={};
    int swap1,i=0;
    //Open file to store random number array
    output.open("random.txt");
    //Fill array
    filAray(array,pFilRow);
```

```
   cout<<"You find the dragon asleep. He wakes up to your presence. However, he says he has two games for
you.\nThe first game\nIf you can guess his number,"
        " then he will let you live and promise never to return to your kingdom. If you lose, he will kill you."<<endl;
   cout<<"The dragon says 'I am thinking of a number between one and ten. What is it?\nYou get 4
guesses.'"<<endl;
   cin>>dgnans;
   //Loop for guessing the dragon's number
   while(dgnans!=dgnnum&&guess<=4){
      if(dgnans<dgnnum)cout<<"Too low!"<<endl;
      if(dgnans>dgnnum)cout<<"Too high!"<<endl;
      cin>>dgnans;
      guess++;
   }
   //If user guessed number in 4 tries or less
   if(guess<=4)
      cout<<"'You guessed my number.'"<<endl;
   //If it took more than 4 tries
   else if(guess>4){
      cout<<"'You did not guess my number. You will die!'"<<endl;
      //If user has less than 4 lifepoints
      if(lifepts<4){
         cout<<"The dragon took the remainder of your life points. You died. GAME OVER."<<endl;
         //End program
         return 0;
      }
      //If user has at least 4 lifepoints
      else{
         cout<<"You survived the dragon's attack, but at the cost of 3 lifepoints."<<endl;
         //Take away 3 lifepoints
         lifepts=lifepts-3;
      }
   }
   cout<<"'Now for the second and final test.'"<<endl;
   cout<<"I will present a block of numbers to you and you must sort them in order. Try to do so in the least amount
of swaps possible."<<endl;
   cout<<"\nPOSITION   0 1 2 3 4 5 6 7 8 9 \n========================================\n";
   //Read file
   fileAry(array,pFilRow,perLine);
   //Print the random number array from the file
   prntfil(array,pFilRow,perLine);
   cout<<endl<<"Which would you like to switch it for? Enter both numbers you wish to switch with by inputing
the corresponding"
        " position numbers above the desired unsorted numbers."<<endl;
   cout<<"Only enter the top position numbers that correspond to the unsorted number below them."<<endl;
   //Loop to keep switching/sorting numbers
   while(ans!='y'&&ans!='Y'){
      cin>>swap1>>i;
      swap(array[i],array[swap1]);
      cout<<"\nPOSITION   0 1 2 3 4 5 6 7 8 9 \n========================================\n";
      prntAry(array,pFilRow,perLine);
      cout<<"Are you done? Enter 'y' for Yes. Enter any other key to continue sorting. Then enter your next number
to sort."<<endl;
      //Prompt to end loop
      cin>>ans;
   }
   cout<<"Ok. Now you are done. I'll compare my answer with yours."<<endl;
```

```
    ans='N';
    //Loop used to check if user correctly sorted the number array
    for(int i=0;i<pFilRow-1;i++){
        //If first number is smaller than second number
        if(array[i]>array[i+1])
            ans='Y';
    }
    cout<<"This is how it should have been sorted."<<endl;
    //Read array from file
    fileAry(array,pFilRow,perLine);
    //Use MarkSort to correctly sort all of the numbers.
    markSrt(array,pFilRow);
    //Print the correct sorted array
    prntAry(array,pFilRow,perLine);
    //Output if user made error when sorting
    if(ans=='Y'){
        cout<<"You made a mistake."<<endl;
        cout<<"Prepare to die!"<<endl;
        //If lifepoints < 4
        if(lifepts<4){
            cout<<"You lose all of your remaining lifepoints. GAME OVER"<<endl;
            return 0;
        }
        else{
            cout<<"You survive the attack. But at the cost of one lifepoint."<<endl;
            lifepts=lifepts-1;
        }
    }
    else
        cout<<"You are correct!"<<endl;
    //If user won at least one of the mini games.
    if(guess<=4||ans!='Y')
        cout<<"You have beaten at least one of my games. I will leave your kingdom and never return."<<endl;
    output.close();
    //Exit
    return lifepts;
}
int dg21(string name,short lifepts){
    ofstream output;
    cout<<"You enter his lair, but the dragon is ready for you and attacks you as soon as it sees you."<<endl;
    //If user has less than 4 lifepoints
    if(lifepts<4){
        cout<<"The dragon took the remainder of your life points. You died. GAME OVER."<<endl;
        //End program
        return 0;
    }
    //If user has at least 4 lifepoints
    else{
        cout<<"You survived the dragon's attack, but at the cost of 3 lifepoints. While the dragon stops to prepare for a
"
                "second attack, you manage to slay the dragon.\nIt is dead!"<<endl;
        //Take away 3 lifepoints
        lifepts=lifepts-3;
    }
    //Exit
    return lifepts;
```

```
}
void filAray(int a[],int n){
   //Loop to generate random numbers from 1 to 10
   for(int i=0;i<n;i++){
      a[i]=rand()%10+1;
   }
   //Exit
}
void prntAry(const int a[],int n,int perLine){
   //Print the array
   cout<<endl<<"           ";
   for(int i=0;i<n;i++){
      //If number is single digit, use a 2 space gap between numbers.
      if(a[i]<10)
         cout<<setw(1)<<a[i]<<"   ";
      //Else use one space gap
      else
         cout<<setw(2)<<a[i]<<"  ";
      if(i%perLine==(perLine-1))cout<<endl;
   }
   cout<<endl;
   //Exit
}
void swap(int &a,int &b){
   //Swap two desired numbers
   if(a==b)
      //If you don't want to swap numbers, or if the numbers are equal to each other
      //      then do nothing
      a=a,b=b;
   else{
      //Swap
      a=a^b;
      b=a^b;
      a=a^b;
   }
   //Exit
}
void fileAry(const int a[],int n,int perLine){
   ofstream output;
   //Open text file
   output.open("random.txt");
   for(int i=0;i<n;i++){
      if(a[i]<10)
         output<<setw(1)<<a[i]<<"   ";
      else
         output<<setw(2)<<a[i]<<"  ";
      if(i%perLine==(perLine-1))output<<endl;
   }
   //Close text file
   output.close();
   //Exit
}
void prntfil(const int a[],int n,int perLine){
   //Read numbers from a file
   ifstream input ("random.txt");
   //Read entire line of numbers as a string
```

```
    string line;
    getline(input,line);
    cout<<endl<<"           ";
    cout<<line;
    cout<<endl;
    //Exit
}
void markSrt(int  a[],int n){
    //Apply swap smallest from beginning to end
    for(int i=0;i<n-1;i++){
        //Swap elements at the top with smallest
        for(int j=i+1;j<n;j++){
            //Swap numbers with smaller ones
            if(a[i]>a[j]){
                int temp=a[i];
                a[i]=a[j];
                a[j]=temp;
            }
        }
    }
    //Exit
}
```