**Project 1 (v2)**

**<LeepFrog Mathematics Game>**

**CSC-5/46024**

**Name: Megan Varner**

**Date: 07/30/14**

**Introduction**

Title: LeepFrog Mathmatics

My game is LeepFrog Mathematics that I grew up playing on a tablet and absolutely loved. The game comes in an arcade like structure and the first game within the game is called “guess the number I’m thinking of,” which is a game of chance. It is used on long car rides and to entertain young children. Someone [known as the chief in the game] will think of a number and not tell anyone. The player will then try and guess the number the chief is thinking of. If done fair, the chief will usually tell a bystander what the number is, to make sure he or she does not change his/her mind mid-game. But as the code is written the computer will not change its guess therefore it is still following the rules. The point of the game is to try and get the least amount of guesses and beat your personal best record. On your own you make a score card and record the amount of guesses it took you each time and then you can just keep playing and recording until you’re tired. You can return to the same score card days later and play again and record the number of guesses each time till you guessed the correct number as well. This game is meant to pass time, there is no strategy and you try and beat your own best record of least amount of guesses till you got the correct number. The next game within LeepFrog Mathematics is an addition game. You try and do the random addition equation and when I played, we use to get treats if we won from our parents. Like the addition game, the subtraction game, is the same concept only with a random subtraction equation and again when we won our parents gave us treats. The next game is a lesson for young children about how numbers are all different, LeepFrog is really good at including these to build a foundation for beginners. The next game is start multiplying and is a game of memorization. LeepFrog includes these for more advance children to learn their multiplication. It’s a chart of all the multiples up to 5. The game is to study the chart on the screen then turn away and write down the chart and see if you got it right. You need to write it down because then it will stick in you memory better and you’ll become better at the game. Once you go it right you moved on to the next game which is start dividing. It has the same concept only with division. The goal is to memorize all your multiplication and division up to 5s. The next game within LeepFrog Mathematics is to try and pick two decimals change them each into percent then add them together on a piece of paper. You then plug it in and check your answer. My mom always helped us when we checked and our answer was wrong. Since we wrote it down she could see our work. This game not only taught us to show our work, but also if we got it, we felt independent because we didn’t need mom, which adds to the fun of the game. The last game within LeepFrog Mathematics is to count by 1s, 2s, 3s, 4s, & 5s. To play you count your twos or threes or any number up to 5s and then see what the computer says is right. You can also play it, like I did, with your brothers each of you count then you press the number and see who is right. Whoever won got bragging rights. This game is usually played on long car rides as well to occupy younger children more advanced in math.

**Summary**

Project size: 650+ lines

The number of variables: about 30

The number of method: 17

My goal was to create an all-age program, that anyone could play and I could use to entertain my young nieces. Therefore, I used the new concepts that we have learned including loops, if/else statements, menus, arrays, functions ect to create a program that is number oriented that they could play and get comfortable with different types of number positive, negative, large, small, ect . The program itself is a work in progress because ultimately I want to make the program into a series of different levels. As of now, there are two player types beginner and expert in elementary math. In the future I hope to add to this and create another program for middle school age students and high school age students. With beginner and expert players as well.

It took about a week to develop the outline the game and the sub-games.

I met a few problems in the process.

I referenced our text book, past homework assignments, and the professor’s online examples to solve them.

**Description**

The main point of this program is how it utilizes numbers and can be used as a learning tool to young children.

**Pseudo Code**

*Initialize*

*Prompt to press 1 or 2 in player options*

*If number one is pressed*

*Prompt to press 1, 2, 3, 4, or 5 in game menu or anything else to exit menu*

*If number 1 is pressed*

*Prompt to guess the random number*

*If number is random number congratulate player*

*Else if number is larger display it’s larger and prompt to guess again*

*Else if number is smaller display it’s smaller and prompt to guess again*

*Play again press P/p or else bool is false and exit sub- game to menu*

*If number 2 is pressed*

*Generate random numbers to be added*

*Prompt to put in answer for random addition equation*

*Output correct answer*

*If answer is correct congratulate player*

*Else answer is wrong and prompt player to play again*

*Play again press P/p or else bool is false and exit sub-game to menu*

*If number 3 is pressed*

*Generate random numbers to be subtracted*

*Prompt to put in answer for random subtraction equation*

*Output correct answer*

*If answer is correct congratulate player*

*Else answer is wrong and prompt player to play again*

*Play again press P/p or else bool is false and exit sub-game to menu*

*If number 4 is pressed*

*Declare and store number*

*Output lesson on numbers*

*If anything else pressed*

*Exit menu*

*If number two is pressed*

*Prompt to press 6, 7, 8, or 9 in game menu or anything else to exit menu*

*If number 6 is pressed*

*Create 5 rows and 5 columns to be outputted*

*Prompt to memorize table on screen*

*Prompt to write the exact table without looking*

*If you were right press Y/y to exit or N/n to try again*

*If number 7 is pressed*

*Create 5 rows and 5 columns to be outputted*

*Prompt to memorize table on screen*

*Prompt to write the exact table without looking*

*If you were right press Y/y to exit or N/n to try again*

*If number 8 is pressed*

*Prompt to pick two decimals, turn into percents, and add together*

*Prompt to enter first decimal*

*Prompt to enter second decimal*

*Tell correct answer*

*Play again press P/p or else bool is false and exit sub- game to menu*

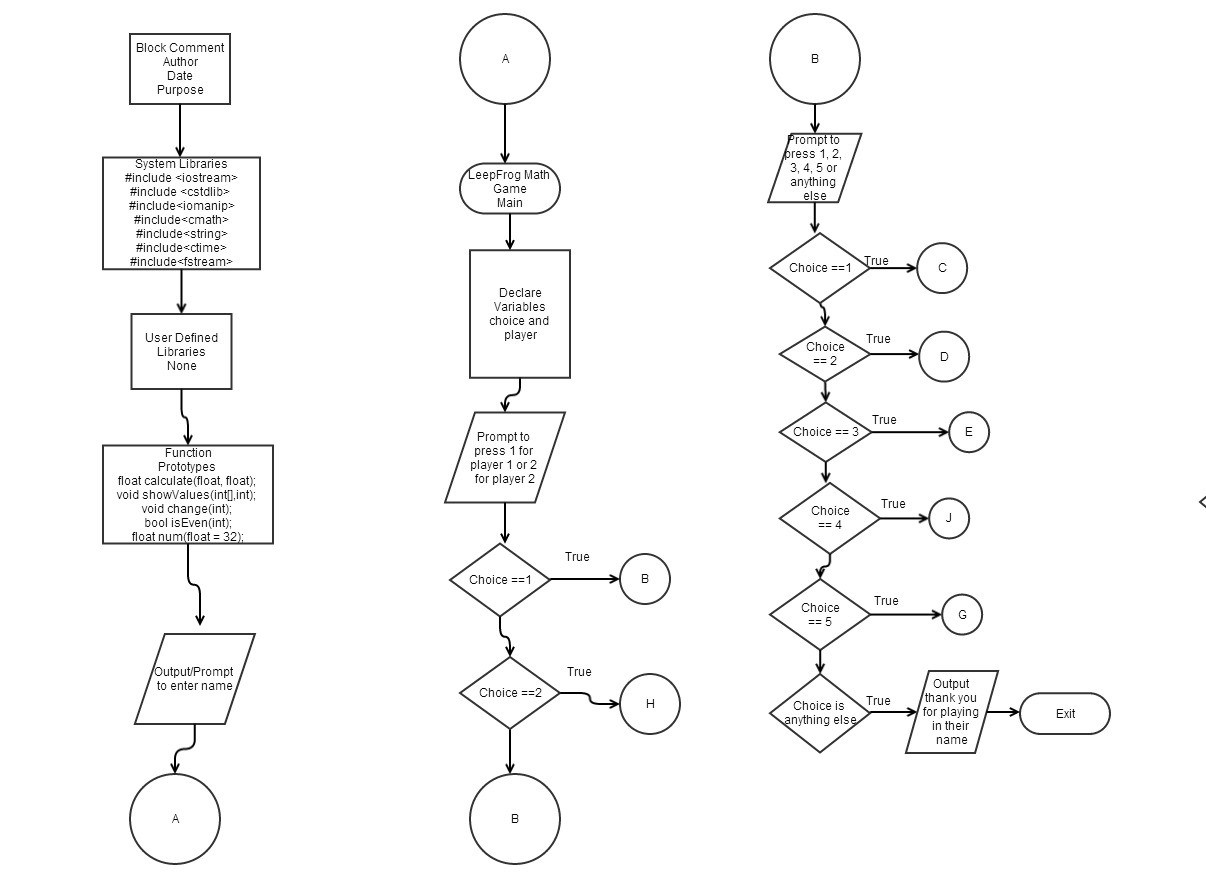
*If number 9 is pressed*

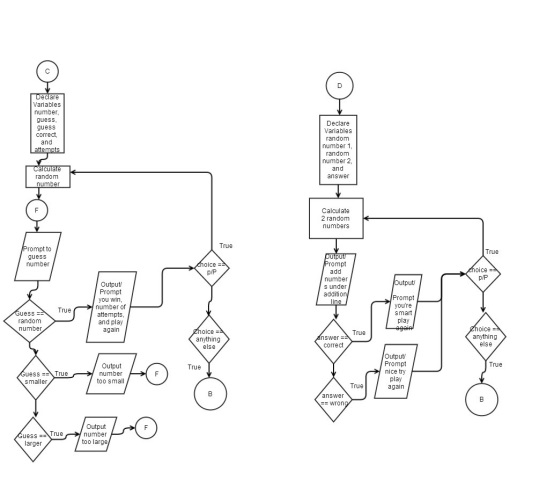
*Prompt to enter 1, 2, 3, 4, or 5*

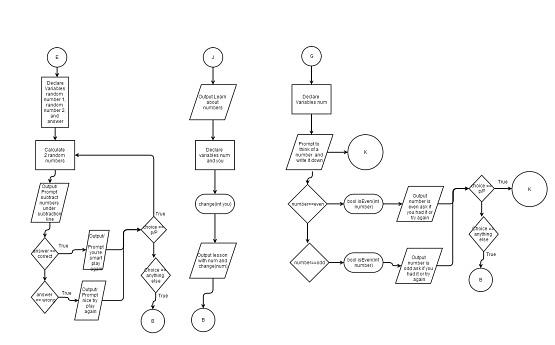
*Output array of counts by number entered*

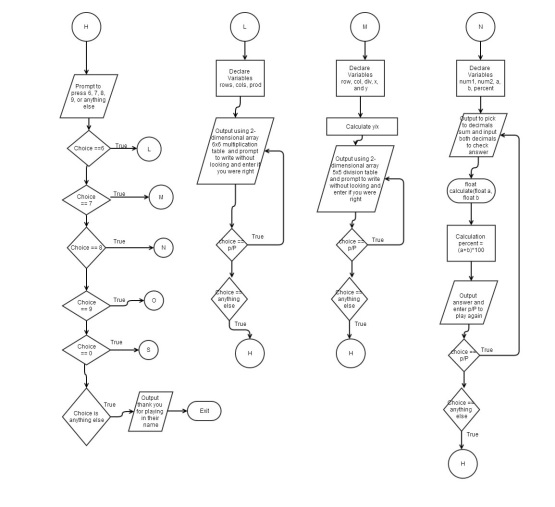
*Play again press P/p or else bool is false and exit sub- game to menu*

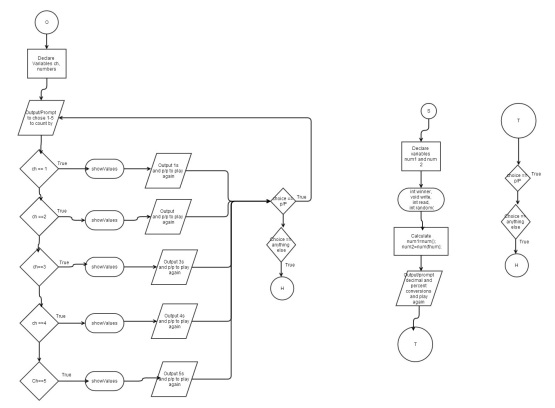
**Project 1 v2 – LeepFrog Mathematics Game FlowChart:**

****









**Major Variables**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Variable Name** | **Description** | **Location** |
| Char | Choice[] [] | Choose sub-game in menu | Line 20 |
| Integer | number[] | Random number in head to guess | Line 48 |
|  | input\_number[] | Player’s guess | Line 51, Line 70 |
|  | attempts\_count[] | Counts the number of guesses a player enters before guessing correctly | Line 52, Line 82 |
| String Choice | choice | Press P/p to play again | Line 97, Line 150, Line 200 |
| Integer | ran | Generates random number | Line 119, Line 168 |
|  | num1 | 1st random number generated for addition /subtraction equation | Line 123, Line172 |
|  | num2 | 2nd random number generated for addition /subtraction equation | Line 124, Line 173 |
|  | num3 | Answer to addition/subtraction equation | Line 133, Line 182 |
| Loop | do{}while() | Loops the menu | Line 24, Line 220 |
| Bool | play\_again | Repeat game from start if true or return to menu if false | Line 21, Line 45, Line 105, Line 116, Line 158, Line 165, Line 208 |
|  | is\_guess\_correct | Guess random number and output while false | Line 56 |
| Counts | attempts\_counts++ | Counts number of attempts made until random number is guessed | Line 82 |
| Menu | 1 | Guess the number I’m thinking of | Line 31 |
|  | 2 | Add with me | Line 32 |
|  | 3 | Subtract with me | Line 34 |
|  | 4 | Lesson on Numbers | Line 61 |
|  | 5 | Even or Odd | Line 62 |
|  | 6 | Start Multiplying | Line 329 |
|  | 7 | Start Dividing | Line 330 |
|  | 8 | Sum the Percents | Line 331 |
|  | 9 | Count by 1s…5s | Line 332 |
| 1 Dimensional Array | mystring | Lets user input name their name and at the end outputs it with a sentence | Line 40-41, Line 537 |
| 2 Dimensional array | Int prod [rows][cols] = {0} | Creates 6x6 multiplication table | Line 348-350 |
|  | float div [row][col] = {0} | Creates 6x6 division table | Line 397-399 |
| Pass Array Between Function | showValues (numbers, Array\_size); | Creates 1, 2, 3, 4, or 5 number sequence for counting | Line 499, 506, 513, 520, and 527 |
| Return Bool Statement | Return stat | Tells if number is even or odd | Line 569-576 |
| Read/write to file | write(sIds,actSize) | Reads/writes the winner when you exit game | Line 594 |
| Defaulted values | num1=num();  num2=num(fnum); | Default parameter call for dec=0 | Line 563 |
| Pass by value | void change(int you | Output 57 when called for in code | Line 264, 624 |
| Return primitive data | return percent; | Allows for output of percent | Line 612 |
|  | return dec/100; | Allows for output of decimal | Line 644 |
|  | return rand()%n; | Allows for output of random number | Line 677 |
| Players | player | Seperates sub-games for beginners and experts | Line 54 |
| functions | int random(int); | Code to find winner randomly | Line 21 |
|  | float calculate(float, float); | Code to calculate percent | Line 27 |
|  | int winner(int [],int); | Code to find winner randomly | Line 22 |
|  | int read(char [],int []); | Code to find winner randomly | Line 23 |

**Reference**

1. textbook

2. https://github.com/mkvarner/mv2504740/tree/master/Hmwk

3. https://github.com/Riverside-City-College-Computer-

Science/Summer14\_CSC5\_46024/tree/master/ml1150258/Examples

**Program**

/\*

\* File: main.cpp

\* Author: Megan Varner

\* Created on July 26, 2014, 9:16 AM

\* Project two game includes functions and arrays

\*/

//Libraries

#include <iostream>

#include <cstdlib>

#include<iomanip>

#include<cmath>

#include<string>

#include<ctime>

#include<fstream>

using namespace std;

//Function winner

//Function Prototype

int random(int);

int winner(int [],int);

int read(char [],int []);

void write(int [],int);

//Function problem 8

float calculate(float, float);

//Function problem 9

void showValues(int[],int);

//Function Problem 4

void change(int);

//Function Problem 5

bool isEven(int);

//Function problem 0

float num(float = 32);

int main(int argc, char\*\* argv)

{

// LeepFrog Mathematics – Game

//Declare Variables

int player;

char choice;

bool play\_again = true;

string mystring[1];

cout<<"What is your name?\n";

getline(cin,mystring[0]);

cout<<"Enter 1 for Beginner Player or 2 for Expert Player: \n";

cin>>player;

//Beginner Player

if (player == 1)

{

//Loop until choice is not in the menu selection

do{

//Start

cout<<"\n";

cout<<"Welcome to my arcade!\n";

cout<<"Lets PLAY!\n";

cout<<"\n";

//Output the menu and input the choice

cout<<"Type 1 to play...guess the number I'm thinking of"<<endl;

cout<<"Type 2 to play...Add with me"<<endl;

cout<<"Type 3 to play...Subtract with me"<<endl;

cout<<"Type 4 to play...Lesson on Numbers"<<endl;

cout<<"Type 5 to play...Even or Odd"<<endl;

cout<<"Type anything else to exit menu"<<endl;

cin>>choice;

play\_again=true;

//Place problem solutions in the switch statement

cout<<endl;

switch(choice){

//guess the number I'm thinking of

case '1':{

cout<<"Guess the number I'm thinking of!!!!!"<<endl<<endl;

cout<<"\n";

while(play\_again == true)

{

srand(time(NULL));

int number = rand() % 56 + 1;

//stores random number between 1-56

bool is\_guess\_correct = false;

int input\_number;

int attempts\_count = 1;

// number of attempts player has made

//loops until player enters correct number

while(is\_guess\_correct == false)

{

if(attempts\_count == 1)

{

cout <<"Lets play!\n";

cout << "I'm thinking of a number 1-56.... ";

cout <<"what is it?\n";

}

else

{

cout << "Guess Again: ";

}

cin >> input\_number;

if(input\_number == number)

{

//Player wins!

cout <<"\n";

cout << "You guessed the number in my head in " ;

cout << attempts\_count << " attempts" << ". Now lets play again!"<<endl;

is\_guess\_correct = true;

//stops loop if true

}

else

{

//Player needs to keep guessing

attempts\_count++;

// attempts

if(input\_number < number)

//if player enters number less than random number

{

cout << "Your guess is smaller than the number I'm thinking of." << endl;

}

else

//if player enters number greater than random number

{

cout << "Your guess is greater than the number I'm thinking of." << endl;

}

}

}

//Play again?

string choice;

cout <<"\n";

cout << "Press P to play again or any other key to exit this game";

cout <<" and return to main menu:";

cin >> choice;

//player chooses any other key than P/p it will exit

if(choice != "P" && choice != "p")

{

play\_again = false;

}

}

break;

}

//Add with me

case '2':{

while(play\_again == true)

{

srand(time(0));

const int ran = 999;

int num1, num2, num3, answer;

//Generates random number

num1 = 1 + rand() % ran;

num2 = 1 + rand() % ran;

//outputs the addition problem

cout << "Lets get smart! Add the numbers and then enter your answer...\n";

cout<<"\n";

cout << setw(5) << num1 << endl;

cout << "+ " << setw(3) << num2 << endl;

cout << "\_\_\_\_\_" << endl;

cin >> answer;

num3 = num1 + num2;

std::cin.get();

cout<<"\n";

cout << num3 << " is the correct answer, ";

//If answer is correct outputs

if (answer == num3)

{

cout << "You're so smart!" << endl;

}

//If answer is wrong outputs

else

{

cout << "Nice try. Want to play again?" << endl;}

string choice;

cout <<"\n";

cout << "Press P to play again or any other key to exit this game";

cout <<" and return to main menu:";

cin >> choice;

//player chooses any other key than P/p it will exit

if(choice != "P" && choice != "p")

{

play\_again = false;

}}

break;

}

//Subtract with me

case '3':{

while(play\_again == true)

{

srand(time(0));

const int ran = 999;

int num1, num2, num3, answer;

//generates random number

num1 = 1 + rand() % ran;

num2 = 1 + rand() % ran;

//outputs subtraction problem

cout << "Lets get smart! Add the numbers and then enter your answer...\n";

cout<<"\n";

cout << setw(5) << num1 << endl;

cout << "- " << setw(3) << num2 << endl;

cout << "\_\_\_\_\_" << endl;

cin >> answer;

num3 = num1 - num2;

std::cin.get();

cout<<"\n";

cout << num3 << " is the correct answer, ";

//outputs if answer is correct

if (answer == num3)

{

cout << "You're so smart!" << endl;

cout<< "Want to go another round?\n";

}

//outputs if answer is wrong

else

{

cout << "Nice try. Want to play again?" << endl;}

string choice;

cout <<"\n";

cout << "Press P to play again or any other key to exit this game";

cout <<" and return to main menu:";

cin >> choice;

//player chooses any other key than P/p it will exit

if(choice != "P" && choice != "p")

{

play\_again = false;

}}

break;

}

//Lesson on Numbers

case '4':{

cout<<"Learn About Numbers!!!!"<<endl;

cout<<endl;

int num = 56;

cout<<num<<" is not the same as"<<endl;

change(num);

cout<<"Because "<<num<<" can only be "<<num<<".\n";

cout<<endl;

cout<<"Each are different quantities so they can't be the same.\n";

cout<<"Numbers are fun! So lets play another game!!!!!!\n";

break;

}

//Even or Odd

case '5':{

int num;

cout<<"See if you know your evens and odds!";

cout<<endl;

cout<<"Think of a number and then write on\n";

cout<<"a piece of paper even or odd for each attempt\n";

while(play\_again == true)

{

cout<<"Write it down\n";

cout<<"Now enter your number: \n";

cin>>num;

if (isEven(num)){

cout<<endl;

cout<<num<<" is even\n";

cout<<"Were you right or wrong?\n";

}

else{

cout<<endl;

cout<<num<<" is odd\n";

cout<<"Were you right?\n";

}

//Play again?

string choice;

cout <<"\n";

cout << "Press Y for yes or N for no and try again: ";

cin >> choice;

cout<<endl;

//player chooses any other key than P/p it will exit

if(choice != "N" && choice != "n")

{

play\_again = false;

}}

break;

}

//exit menu when 1-6 are not entered

default:{

cout<<"Exit Menu"<<endl<<endl;

break;

};

}

}while(choice >= '1' && choice <='5'); //ends loop i.e. arcade

}

//Expert Player

else

{

//Loop until choice is not in the menu selection

do{

//Start

cout<<"\n";

cout<<"Welcome to my arcade!\n";

cout<<"Lets PLAY!\n";

cout<<"\n";

//Output the menu and input the choice

cout<<"Type 6 to play...Start Multiplying"<<endl;

cout<<"Type 7 to play...Start Dividing"<<endl;

cout<<"Type 8 to play...Sum the Percents"<<endl;

cout<<"Type 9 to play...Count by 1s, 2s, 3s, 4s, & 5s"<<endl;

cout<<"Type 0 for extra credit! See if you can do conversions!\n";

cout<<"Type anything else to exit menu"<<endl;

cin>>choice;

play\_again=true;

//Place problem solutions in the switch statement

cout<<endl;

switch(choice){

//Start Multiplying

case '6':{

while(play\_again == true)

{

//2-dimensional array

//int anArray[5][4];

// 6x6 array

const int rows = 6;

const int cols = 6;

int prod[rows ][cols ] = { 0 };

cout<<"Try and memorize this multiplication table!\n";

cout<<"Study the table on the screen...\n";

cout<<endl;

// Create table

for (int nRow = 0; nRow < rows; nRow++)

for (int nCol = 0; nCol < cols; nCol++)

//multiplication calculation

prod[nRow][nCol] = nRow \* nCol;

// output the table

for (int nRow = 1; nRow < rows; nRow++)

{

for (int nCol = 1; nCol < cols; nCol++)

cout << prod[nRow][nCol] << "\t";

cout << "\n";

}

cout<<"\nNow try and write the exact table without looking!\n";

cout<<endl;

cout<<"Were you right?\n";

//Play again?

string choice;

cout << "Press Y for yes or N to output the table again to study more: ";

cin >> choice;

cout<<endl;

//player chooses any other key than P/p it will exit

if(choice != "N" && choice != "n")

{

play\_again = false;

}}

break;

}

//Start Dividing

case '7':{

//2-dimensional array

//float anArray[5][4];

// 6x6 array

const int row = 5;

const int col = 5;

float div[row][col] = { 0 };

float x=0;

float y=0;

// Create table

for (int nRow = 0; nRow < row; nRow++){

x++;

for (int nCol = 0; nCol < col; nCol++){

y++;

cout<<"x:"<<x<<"divided by y:"<<y<<" = "<<y/x<<"\n";

//Division calculation

div[nRow][nCol] = y/x;

}

y=0;

}

// output the table

int rowstart=1;

cout<<"0 1 2 3 4 5\n";

for (int nRow = 0; nRow < row; nRow++)

{

cout<<rowstart<<" ";

rowstart++;

for (int nCol = 0; nCol < col; nCol++){

cout << div[nRow][nCol] << "\t";

}

cout << "\n";

}

break;

}

//Sum the Percents

case '8':{

float num1, num2;

cout<<"The challenge:\n";

cout<<"Pick two decimals and turn them into percents ";

cout<<"and add together on paper...\n";

cout<<"Then check your solution by solving on here! Ready, set, go!\n";

while(play\_again == true)

{

cout<<endl;

cout << "Enter first number in decimal form:\n";

cin >> num1;

while (num1<=0){

cout << "Enter first number in decimal form:\n";

cin >> num1;

}

cout << "Enter second number in decimal form to be added to first:\n";

cin >>num2;

while (num2<=0){

cout << "Enter second number in decimal form to be added to first:\n";

cin >> num2;

}

cout << "The sum is %" << calculate(num1, num2)<<endl;

//Play again?

string choice;

cout <<"\n";

cout << "Press P to play again or any other key to exit this game";

cout <<" and return to main menu:";

cin >> choice;

//player chooses any other key than P/p it will exit

if(choice != "P" && choice != "p")

{

play\_again = false;

}

}

break;

}

//Count by 1s, 2s, 3s, 4s, & 5s

case '9':{

cout<<"Lets count by 1s, 2s, 3s, 4s, & 5s!!!!!"<<endl<<endl;

while(play\_again == true)

{

int ch;

cout<<"Enter what to count by 1, 2, 3, 4, or 5:\n";

cin >>ch;

if (ch == 1) {

const int Array\_size = 10;

int numbers [Array\_size] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

showValues (numbers, Array\_size);

}

if (ch == 2) {

const int Array\_size = 10;

int numbers [Array\_size] = {2, 4, 6, 8, 10, 12, 14, 16, 18, 20};

showValues (numbers, Array\_size);

}

if (ch == 3) {

const int Array\_size = 10;

int numbers [Array\_size] = {3, 6, 9, 12, 15, 18, 21, 24, 27, 30};

showValues (numbers, Array\_size);

}

if (ch == 4) {

const int Array\_size = 10;

int numbers [Array\_size] = {4, 8, 12, 16, 20, 24, 28, 32, 36, 40};

showValues (numbers, Array\_size);

}

if (ch == 5) {

const int Array\_size = 10;

int numbers [Array\_size] = {5, 10, 15, 20, 25, 30, 35, 40, 45, 50};

showValues (numbers, Array\_size);

}

if (ch>5){

cout<<"Invalid Entry\n";

}

string choice;

cout <<"\n";

cout << "Press P to play again or any other key to exit this game";

cout <<" and return to main menu:";

cin >> choice;

//player chooses any other key than P/p it will exit

if(choice != "P" && choice != "p")

{

play\_again = false;

}}

break;

}

//Conversions

case '0':{

cout<<"Lets do conversions!!!!!"<<endl<<endl;

//Set the random number seed

srand(static\_cast<unsigned int>(time(0)));

while(play\_again == true)

{

//Declare a decimal

float fnum=rand()%100+0;//number between [0,100]

//declare variables

float num1, num2;

//Calculate the conversions

num1=num();//Default parameter call for dec=0;

num2=num(fnum);

cout<<"\n"<<fnum<<"% converts to => "<<num2<<" and\n";

cout<<num2<<" converts to => "<<fnum<<"%\n";

cout<<"Now show your work and explain how that works!\n";

string choice;

cout <<"\n";

cout << "Press P to play again or any other key to exit this game";

cout <<" and return to main menu:\n";

cin >> choice;

//player chooses any other key than P/p it will exit

if(choice != "P" && choice != "p")

{

play\_again = false;

}}

break;

}

}

}while(choice >= '4' && choice <='9' || choice == '0'); //ends loop i.e. arcade

}

//Declare variables and initialize

char fName[]="\n";

const int SIZE=35;

int sIds[SIZE];

srand(static\_cast<unsigned int>(time(0)));

//Read in the data

int actSize=read(fName,sIds);

//Print the data in the file

write(sIds,actSize);

//Choose the Winner

cout<<"The winner is....."

<<mystring[0]<<endl;

cout<<"\nCome play again "<<mystring[0]<<" and we can learn some more!\n";

//Exit stage right!

return 0;

}

//Function Definitions 8

float calculate(float a, float b)

{

float percent;

//Convert and sum percents

percent = (a+b) \* 100;

return percent;

}

//Function Definitions 9

void showValues(int nums[], int size)

{

for (int index = 0; index < size; index++)

cout<<nums[index]<<" ";

cout<<endl;

}

//Function Definitions 4

void change(int you){

you = 57;

cout<<you<<endl;

}

//Function Definitions 5

bool isEven(int number){

bool stat;

if(number % 2 == 0)

stat = true;

else

stat = false;

return stat;

}

//Function definition 0

float num(float dec){

//Process the input and return the value

return dec/100;

}

//Function definition winner

int winner(int sId[],int n){

return sId[random(n)];

}

void write(int sId[],int size){

cout<<endl;

cout<<"GUESS WHAT?! YOU'RE AWESOME!"<<endl;

cout<<endl;

}

int read(char fName[],int sId[]){

//Declare variables

int cnt=0;

ifstream input;

//Open the file

input.open(fName);

//Read the data

while(input){

input>>sId[cnt++];

}

//close the file

input.close();

//exit

return --cnt;

}

int random(int n){

return rand()%n;

}