```
Script started on 2021-04-15 18:24:11-0500
m_sadaf1@ares:~$ pwd
/home/students/m_sadaf1
m_sadaf1@ares:~$ cat louie.info
Name: Madiha Sadaf
Class: CSC122 W01

Lab: A little to the left, Louie!

Level: 2

Description:

This program creates a table for the user's choice of
basic math operations. It also allows the user to choose
the size of the table.

m_sadaf1@ares:~$ cat louie.tpq
Thought Provoking Questions:

1) To easily determine the widest that the columns need
   to be, I found out the length of the columns that
   turned out to be greater than 60 characters, which
   would occur at approximately 20 columns.

2) Once the width of the column is known, the table will
   fit perfectly since the output will be eventually in
   the shape of a square. So the height and width will
   coordinate with the sqaure shape of the screen used
   in the program.

3) By using #include <iomanip> and setw() within the
   iomanip library, we can line up the columns to be
   nice and neat.

4) By using cases set to choose an outcome based on the
   user input inside the while/switch statement, I can
   get my menu to accept both character and numeric input.

5) No, I did not need any flags to line up my columns
   neatly.



m_sadaf1@ares:~$ cat louie.cpp
#include <iostream>
#include <iomanip>
#include <string>

using namespace std;

int get_num(int value)
{
    int total = 0;
    if(value <= 0)
    {
        ++total;
    }
    while(value != 0)
    {
        value /= 10;
        ++total;
    }
    return total;
}

int get_value(int i, int j, char oper)
{
    switch(oper)
    {
        case '+':
            return i+j;
        case '-':
            return i-j;
        case '*':
            return i*j;
        case '/':
            return i/j;
        case '%':
```

```
            return i%j;
        default:
            return 0;
    }
}

int get_max(int n, char ch)
{
    int max= 0;
    for(int i = 1; i <=n; ++i)
    {
        for(int j = 1; j <= n; ++j)
        {
            int n = get_num(get_value(i, j, ch));
            if(n > max)
            {
                max = n;
            }
        }
    }
    return max + 1;
}

void table(int n, char oper)
{
    int max = get_max(n, oper);
    cout << setw(max) << oper << "|";
    for(int i = 1; i <= n; ++i)
    {
        cout << setw(max) << i;
    }
    cout << endl;
    for(int i = 0; i < max; ++i)
    {
        cout << "-";
    }
    cout << "+";
    for(int i = 0; i < n * max; ++i)
    {
        cout << "-";
    }
    cout << endl;
    for(int i = 1; i <=n; ++i)
    {
        cout << setw(max) << i << "|";
        for(int j = 1; j <= n; ++j)
        {
            cout << setw(max) << get_value(i, j, oper);
        }
        cout << endl;
    }
    cout << endl;
}

char get_choice()
{
    cout << "Table Menu";
    cout << "1) Addition table";
    cout << "2) Multiplication table";
    cout << "3) Subtraction table";
    cout << "4) Division table";
    cout << "5) Remainder table";
    cout << "6) Quit";
    cout << "Choice: ";

    char choice;
    cin >> choice;
    return choice;
}

int get_size(string type)
{
    int size;
    while(true)
    {
        cout << "What size should the " << type <<
                " table be? ";

        cin >> size;
        if(size <= 0)
```

```
        {
            cout << "I'm sorry, " << size <<
                    " would be too small to print on the "
                    " screen...";
        }
        else if(size >= 20)
        {
            cout << "I'm sorry, " << size <<
                    " would be too large to print on the "
                    " screen...";
        }
        else
        {
            cout << "Thank you...calculating...";
            return size;
        }
    }
}

int main()
{
    cout << "\t\tWelcome to the Math Table Program!!!";
    char choice;
    int size;
    while(true)
    {
        choice = get_choice();
        switch(choice)
        {
            case 'a':
            case 'A':
            case '1':
                size = get_size("addition");
                table(size, '+');
                break;
            case 'm':
            case 'M':
            case '2':
                size = get_size("multiplication");
                table(size, '*');
                break;
            case 's':
            case 'S':
            case '3':
                size = get_size("subtraction");
                table(size, '-');
                break;
            case 'd':
            case 'D':
            case '4':
                size = get_size("division");
                table(size, '/');
                break;
            case 'r':
            case 'R':
            case '5':
                size = get_size("remainder");
                table(size, '%');
                break;
            case 'q':
            case 'Q':
            case '6':

                cout << "Thank you for using the MTP!!!";
                cout << "Endeavor  to have a auspicious day";
                return 0;
            }
        }
        return 0;
    }

m_sadaf1@ares:~$ CPP louie
louie.cpp***
louie.cpp: In function ‘int get_max(int,
char)’:
louie.cpp:48:17: warning:
declaration of ‘int n’ shadows a parameter
[-Wshadow]
             int n = get_num(get_value(i, j, ch));
                 ^
```

```
louie.cpp:41:17: note: shadowed
declaration is here
 int get_max(int n, char ch)
                 ^



m_sadaf1@ares:~$ exit
exit

Script done on 2021-04-15 18:24:59-0500
```