**Take Home Program #4 – Due on or before March 12, 2019 –**

**Objectives:** while and do while loops

|  |
| --- |
| **Important instructions:**   * *All programs must include comments at the top of your program: your name,* the class name (CSIT 575)*, program name and* ***the program description (purpose of the program).*** * *Copy and paste your* ***program code*** *and* ***output*** *in Part B of each program. Note: Use snipping tool to snip the output.* * *Once it is done, save and submit this word file via Canvas.* |

1. **DisplayASCII.cpp** (while, modulus, single if statement, type casting)

Write a program that uses a loop to display the characters for each ASCII code 32 through 127. Display 16 characters on each line with one space between characters.

Note that ASCII 32 is a blank space, so the first “character” printed on the first output line is a blank.

**Sample run 1**

! " # $ % & ' ( ) \* + , - . /

0 1 2 3 4 5 6 7 8 9 : ; < = > ?

@ A B C D E F G H I J K L M N O

P Q R S T U V W X Y Z [ \ ] ^ \_

` a b c d e f g h i j k l m n o

p q r s t u v w x y z { | } ~ 

**Part B: Copy and paste your program (source) code and the outputs after this line.**

**++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++**

/\* Erik Gonzalez

CO SCI 575

Display ASCII Table.cpp

This is a program that displayers the ASCII table from characters from 32 to 127 while making sure there

are 16 characters per line\*/

#include <iostream>

using namespace std;

int main()

{

int val, // Value

letters = 32;

for (int Row = 0; Row < 6; Row++) // loop to make 6 rows

{

val = letters + 16; // 16 character on each line

for (int ASCII = letters; ASCII < val; ASCII++)

{

if (ASCII % 128 == 0) // Makes sure every row is organized and outputs correctly.

cout << endl;

cout << static\_cast<char>(ASCII) << " ";

}

letters = val; // Start new row one character than last line

cout << endl;

}

system("pause");

return 0;

}

Output:

! " # $ % & ' ( ) \* + , - . /

0 1 2 3 4 5 6 7 8 9 : ; < = > ?

@ A B C D E F G H I J K L M N O

P Q R S T U V W X Y Z [ \ ] ^ \_

` a b c d e f g h i j k l m n o

p q r s t u v w x y z { | } ~ 

1. **MathTutor.cpp** (do while,switch statements, if else/if trailing else, random generator, stream manipulator)

Write a program that can be used a math tutor for a young student. The program displays a menu allowing the user to select an addition, subtraction, or multiplication problem. The final selection on the menu should let the user quit the program. After user has finished the math problem, the program should display the menu again. This process must be repeat until the user chooses to quit the program. It the select an item not on the menu, the program should print an error message and the display the menu again.

**Notes:** The program should display the two random numbers between 1 – 50.

For subtraction, the second random bust be smaller the first random

For multiplication, the second random generates from 1 – 9.

**Sample run:**

|  |  |
| --- | --- |
|  |  |

**Part B: Copy and paste your program (source) code and the outputs after this line.**

**++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++**

/\*Erik Gonzalez

CO SCI 575

MathTutor.cpp

This program will be used to help tutor math students\*/

#include <iostream>

#include <iomanip>

#include <cstdlib>

#include <ctime>

using namespace std;

int main()

{

int num1,

num2,

choice, // The user's choice of problem

studentAnswer,

correctAnswer;

srand(time(0));

do

{

cout << "\tMenu\n";

cout << "1. Addition problem\n";

cout << "2. Subtraction problem\n";

cout << "3. Multiplication problem\n";

cout << "4. Quit this program\n";

cout << "Enter your choice (1-4): ";

cin >> choice;

// Validate the choice.

if (choice < 1 || choice > 4)

{

cout << "Invalid input. You must enter 1-4\n" << endl;

}

// Produce a problem.

switch (choice)

{

case 1: // Addition problem

num1 = 1 + rand() % 50;

num2 = 1 + rand() % 50;

correctAnswer = num1 + num2;

cout << "\n\n";

cout << " " << setw(4) << num1 << endl;

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

cout << " " << "----" << endl;

cout << " ";

break;

case 2: // Subtraction problem

num1 = 1 + rand() % 50;

num2 = rand() % num1;

while (num2 > num1)

num2 = 1 + rand() % 50;

// Get the correct answer.

correctAnswer = num1 - num2;

// Display the problem.

cout << "\n\n";

cout << " " << setw(4) << num1 << endl;

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

cout << " " << "----" << endl;

cout << " ";

break;

case 3: // Multiplication problem

// Generate two random number

num1 = 1 + rand() % 9;

num2 = 1 + rand() % 9;

correctAnswer = num1 \* num2;

cout << "\n\n";

cout << " " << setw(4) << num1 << endl;

cout << " \*" << setw(3) << num2 << endl;

cout << " " << "----" << endl;

cout << " ";

break;

case 4: // quit the program.

cout << "Thank you for using Math Tutor.\n\n";

break;

}

if (choice >= 1 && choice <= 3)

{

cin >> studentAnswer;

if (studentAnswer == correctAnswer)

cout << "\n\nCongratulations! That's right.\n\n";

else

cout << "\n\nSorry, the correct answer is " << correctAnswer

<< ".\n\n";

}

} while (choice != 4); // Loop again if the choice is not 1, 2, 3, or 4

return 0;

}

**Menu**

**1. Addition problem**

**2. Subtraction problem**

**3. Multiplication problem**

**4. Quit this program**

**Enter your choice (1-4): 2**

**38**

**- 29**

**----**

**46**

**Sorry, the correct answer is 9.**

**Menu**

**1. Addition problem**

**2. Subtraction problem**

**3. Multiplication problem**

**4. Quit this program**

**Enter your choice (1-4): 3**

**8**

**\* 4**

**----**

**32**

**Congratulations! That's right.**

**Menu**

**1. Addition problem**

**2. Subtraction problem**

**3. Multiplication problem**

**4. Quit this program**

**Enter your choice (1-4): 1**

**30**

**+ 33**

**----**

**63**

**Congratulations! That's right.**

**Menu**

**1. Addition problem**

**2. Subtraction problem**

**3. Multiplication problem**

**4. Quit this program**

**Enter your choice (1-4):**