// =======================

// Included: HW 3

// =======================

// HW 3

// =======================

// Christian Falucho

// CMPR 121

// =======================

#include <iostream>

#include <iomanip>

using namespace std;

/\*================ GLOBAL VARIABLES ===================\*/

const int MAX\_HOURS = 23;

const int MAX\_MINUTES = 59;

const int MAX\_SECONDS = 59;

/\*================ GLOBAL VARIABLES ===================\*/

/\*================ STRUCTS ==============================\*/

struct Time{

int hours;

int minutes;

int seconds;

};

/\*================ STRUCTS ==============================\*/

/\*================ FUNCTION PROTOTYPES ==================\*/

void getTime(Time&);

bool isTimeValid(Time&);

void addOneSecond(Time&);

void displayTime(Time&);

void repeatProgram(Time&);

/\*================ FUNCTION PROTOTYPES ==================\*/

/\*

===========================================================

=== MAIN FUNCTION BEGINS ===

===========================================================

\*/

int main() {

Time time;

char answer = 'y';

do

{

system("clear");

cout << "Enter the time in 'military time', (24-hour format),

in the following order: HH:MM:SS, (Hours, Minutes, Seconds)." << endl << endl;

getTime(time);

addOneSecond(time);

displayTime(time);

cout << "Do you want to re-run the program? (Y/N) ";

cin >> answer;

} while (toupper(answer) != 'N');

}

/\*

===========================================================

=== MAIN FUNCTION ENDS ===

===========================================================

\*/

/\*

===========================================================

=== CODE OUTPUT ===

===========================================================

\*/

A screenshot of a computer program

AI-generated content may be incorrect.

* When time input is invalid. User is prompted to input time again.

A screen shot of a computer

AI-generated content may be incorrect.

/\*

===========================================================

=== HELPER FUNCTION BEGINS ===

===========================================================

\*/

void getTime(Time& time){

char answer;

do

{

cout << "Enter hours: ";

cin >> time.hours;

cout << "Enter minutes: ";

cin >> time.minutes;

cout << "Enter seconds: ";

cin >> time.seconds;

} while (isTimeValid(time) != true);

cout << endl;

cout << "Do you want to continue? (Y/N) ";

cin >> answer;

if (toupper(answer) == 'Y')

{

getTime(time);

}

}

// Check if the time inputted is between 0:00:00 to 23:59:59

bool isTimeValid(Time& time){

bool valid = true;

if ((time.hours >= 0) && (time.hours <= MAX\_HOURS) &&

(time.minutes >= 0) && (time.minutes <= MAX\_MINUTES) &&

(time.seconds >= 0) && (time.seconds <= MAX\_SECONDS))

{

return valid;

}

else{

valid = false;

cout << endl;

cout << "Invalid time. Try again." << endl;

return valid;

}

return !valid;

}

// Function to add a second to time

// When time for each unit go pass the max, set to 0.

// Ex. 07:59:59 -> 08:00:00

void addOneSecond(Time& time){

cout << endl;

cout << "After adding one second, the time is ";

time.seconds++;

if (time.seconds > MAX\_SECONDS)

{

time.seconds = 0;

time.minutes++;

if (time.minutes > MAX\_MINUTES)

{

time.minutes = 0;

time.hours++;

if (time.hours > MAX\_HOURS)

{

time.hours = 0;

}

}

}

}

void displayTime(Time& time){

cout.fill('0');

cout << setw(2) << time.hours << ":" << setw(2) << time.minutes << ":" << setw(2) << time.seconds << "." << endl << endl;

}

/\*

===========================================================

=== HELPER FUNCTION ENDS ===

===========================================================

\*/