**Instructor:** Yerasyl Amanbek

**Student:** Amir Yakubov

**Date:** 17.04.2021

**Part 1. Heart-rate Monitor in C (25%)**

**1.** While exercising, you can track your heart-rate to see if it stays within a safe range according to formula:

**- maximum HR** (when exercising): 220 - your age

**- target HR** (when exercising): 50-85% of you maxHR

Create a program in **C** where user can enter his name, date of birth (day, month and year separately). Then calculate age of the person in separate function. Display information on person’s age, maxHR and targetHR­.

Your program should also receive from user array of integers (heart-rates) e.g. {65, 58, 61, 81, …}. Display warning message if heart-rate is below or above targetHR. Provide screenshot of your program

**2.** Record your current heart-rate (explain or show how you tracked your heart rate).

**I have tracked heart rate using** smart watch. I have done some experiments. While doing my homework, that is, sitting at rest, my heart rate is 60-100 bpm. After a light warm-up, the minimum level slightly increased and stayed in the range from 70 to 80 bpm. Then I used more load, after which the pulse became 101-170 bpm.

**3.** Do some warm up(2-5min) and record your heart-rate again. Fill the table below:

|  |  |
| --- | --- |
| **Hear-rate** | **Value** |
| Your targetHR (when exercising) | **101-170 bpm** |
| While doing this assignment | **60-100 bpm** |
| After warm up | **70-80 bpm** |

*Note: when resting (no physical activity) normal heart rate 55-80.*

**4.** Provide codes:

#include <stdio.h>

char firstName[50], lastName[50];

int day, month, year, age;

char fname(char fname[]) {

printf("\nEnter your first name: ");

scanf("%s", fname);

return printf("Your name is %s.", fname);

}

char lname(char lname[]) {

printf("\n\nEnter your last name: ");

scanf("%s", lname);

return printf("Your surname is %s.", lname);

}

int d(int d) {

printf("\n\nEnter your birth day: ");

scanf("%d", &d);

return printf("Your birth day is %d.", d);

}

int m(int m) {

printf("\n\nEnter your birth mounth (in numbers): ");

scanf("%d", &m);

return printf("Your birth mounth is %d.", m);

}

int y(int y) {

printf("\n\nEnter your birth year: ");

scanf("%d", &y);

return printf("Your birth year is %d.\n", y);

}

int Age(int age){

printf("\n\nPlease enter your birth year again: ");

scanf("%d", &year);

age = 2021 - year;

return age;

}

int MaxiumumHeartRate(int age)

{

int maxHR = 220 - age;

return maxHR;

}

int min, max;

void TargetHeartRate(int maxHR)

{

min = (maxHR \* 50)/100;

max = (maxHR \* 85)/100;

if(maxHR < max && max > maxHR) {

printf("\nHigh!");

}

else if(max > min && min <max){

printf("\nYour target heart rate range is from %d", min);

     printf(" to %d", max);

}

else {

printf("\nLow!");

}

}

void displayData(int age)

{

int a = age;

printf("\nHello %s ",firstName);

printf("%s", lastName);

printf("\nYou are ");

printf("%d",a); printf(" years old");

}

int main() {

fname(firstName);

lname(lastName);

d(day);

m(month);

y(year);

int age = Age(age);

int maxHR = MaxiumumHeartRate(age);

displayData(age);

TargetHeartRate(maxHR);

return 0;

}

**Part 2. Temperature convertor in C** **(25%)**

**1.** Write a program that converts Fahrenheit to Celsius and vice versa. Ask the user which converter he wants to use. Ask the user to input some temperature. Covert the temperature and display both temperatures (first is typed temperature).

**2.** Gather at least 7 facts about temperature that actually includes temperature value. E.g: water boils at 100°C or the coldest temperature recorded in Earth: −128.6 °F.

1. In universe temperatures range from about 3,500,000,000 Kelvin (a supernova) to 3 Kelvin (space).

2. Sun is known to be a class G yellow star. The average surface temperature of Sun is 5,600 Kelvin.

3. Absolute zero is the coldest theoretical temperature. It has been defined as zero Kelvin (0 Kelvin) which is found to be equivalent to -273.16 degrees Celsius and -459.69 degrees Fahrenheit.

4. Water boils at 100C

5. Fahrenheit and Celsius are equal at -40 degrees.

6. 57.8 °C (136 °F) is the hottest temperature ever recorded on Earth.

7. −89.2 °C (−128.6 °F) is the coldest temperature ever recorded on Earth.

**3.** Modify the program to also display interesting fact about temperature after showing conversion. Display the fact about temperature that have value closest to the temperature typed by the user. E.g., if user types: 104 °C program displays: Interesting fact: water boils at 100 °C

**4.** Provide codes:

#include<stdio.h>

int main()

{

float fahrenheit, celcius;

int choice;

printf("Enter 1 : Fahrenheit to celcius \nEnter 2 : Celcius to Fahrenheit\n");

scanf("%d", &choice);

if (choice == 1)

{

printf("Enter the Fahrenheit value :\n");

scanf("%f", &fahrenheit);

celcius = (fahrenheit - 32) / 1.8;

if(fahrenheit > 0 && fahrenheit < 200) {

printf("\nInteresting fact: 136 °F is the hottest temperature ever recorded on Earth\n\n");

}

else if(fahrenheit < 0 && fahrenheit > -200 && fahrenheit != -40) {

printf("\nInteresting fact: −128.6 °F is the coldest temperature ever recorded on Earth\n\n");

}

else if (fahrenheit == (-40) ) {

printf("\nInteresting fact: Fahrenheit and Celsius are equal at -40 degrees\n\n");

}

else {

printf("\nIn universe temperatures range from about 3,500,000,000 Kelvin (a supernova) to 3 Kelvin (space)\n\n");

}

printf("%f", fahrenheit);

printf(" degree Fahrenheit\n");

printf(" = \n");

printf("%f", celcius);

printf(" degree celcius\n");

}

else if (choice == 2)

{

printf("Enter the Celcius value :\n");

scanf("%f", &celcius);

fahrenheit = (1.8 \* celcius) + 32;

if(celcius > 0 && celcius <200) {

printf("\nInteresting fact: Water boils at 100°C\n\n");

}

else if(celcius < 0 && celcius > -100 && celcius != -40) {

printf("\nInteresting fact: Fahrenheit and Celsius are equal at -40 degrees\n\n");

}

else if (fahrenheit == (-40) ) {

printf("\nInteresting fact: Fahrenheit and Celsius are equal at -40 degrees\n\n");

}

else {

printf("\nIn universe temperatures range from about 3,500,000,000 Kelvin (a supernova) to 3 Kelvin (space)\n\n");

}

printf("%f", celcius);

printf(" degree celcius\n");

printf(" = \n");

printf("%f", fahrenheit);

printf(" degree Fahrenheit\n");

}

else

{

printf("Please enter a valid input\n");

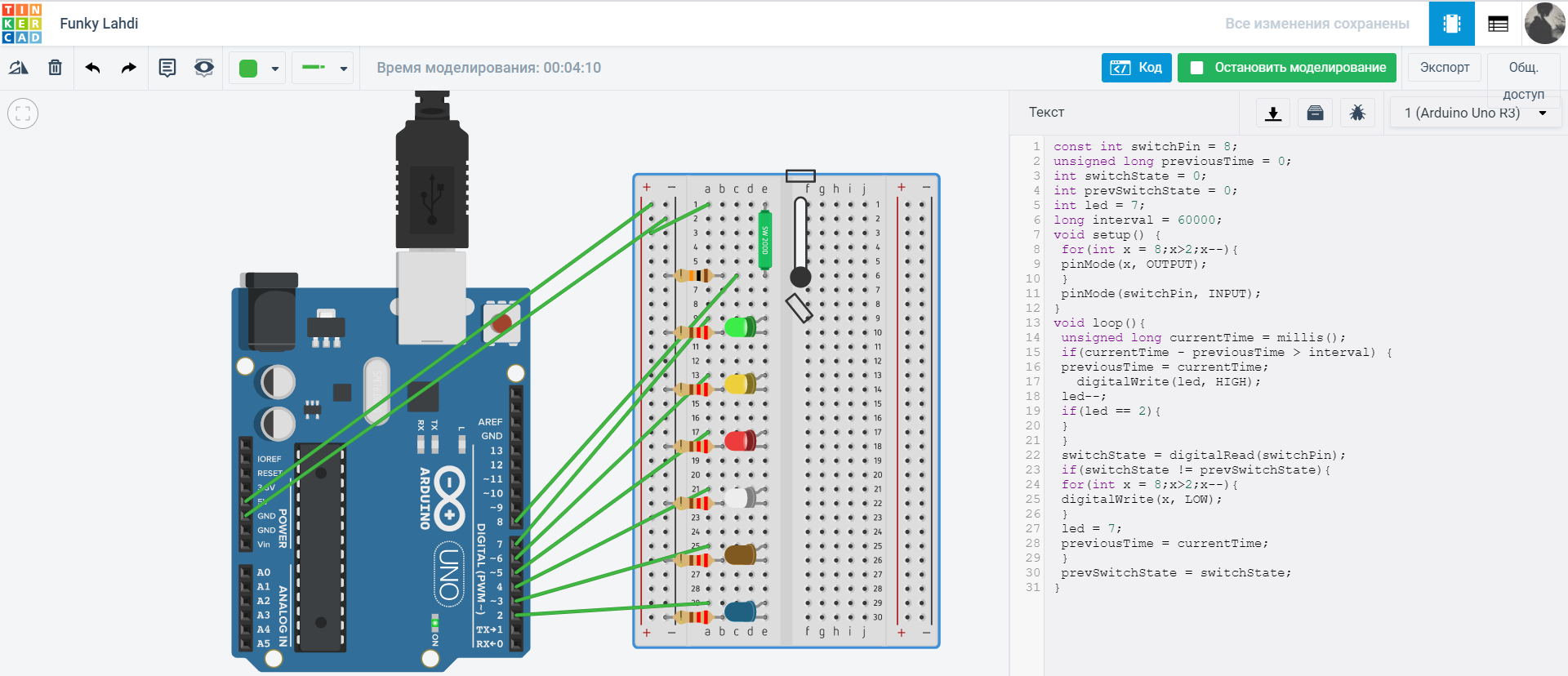
}

return 0;

}

**Part 3. Modified project 8 from Arduino Projects Book.** **(25%)**  
*Build project 8 from Arduino Projects Book in Tinkerсad.  
Modify the project: play with switchState variable to indicate what direction the lights should go.* ***Make real hourglass.***

1. Provide screenshot of the circuit.



1. Explain why we need millis() function?

millis () allows us to access the current timer, returns in milliseconds.

1. In which projects we could use millis() function?

We can use them to make a watch, as soon as the download is complete, the time will start from that moment.

1. In which projects we could use tilt switch?

Using the tilt sensor, you can turn on the light, or start to execute any command.  
  
**5.** Provide codes with comments.

const int switchPin = 8;

unsigned long previousTime = 0; // previousTime indicates the last time an LED was turned on

int switchState = 0;

int prevSwitchState = 0;

int led = 7;

long interval = 600;

void setup() {

for(int x = 8;x > 2;x--){ // condition for turning on the lights from the back side

pinMode(x, OUTPUT);

}

pinMode(switchPin, INPUT);

}

void loop(){

unsigned long currentTime = millis();

if(currentTime - previousTime > interval) {

previousTime = currentTime;

digitalWrite(led, HIGH);

led--;

if(led == 2){

}

}

switchState = digitalRead(switchPin);

if(switchState != prevSwitchState){ // If they are different, turn the LEDs off

for(int x = 8;x > 2;x--){

digitalWrite(x, LOW);

}

led = 7;

previousTime = currentTime;

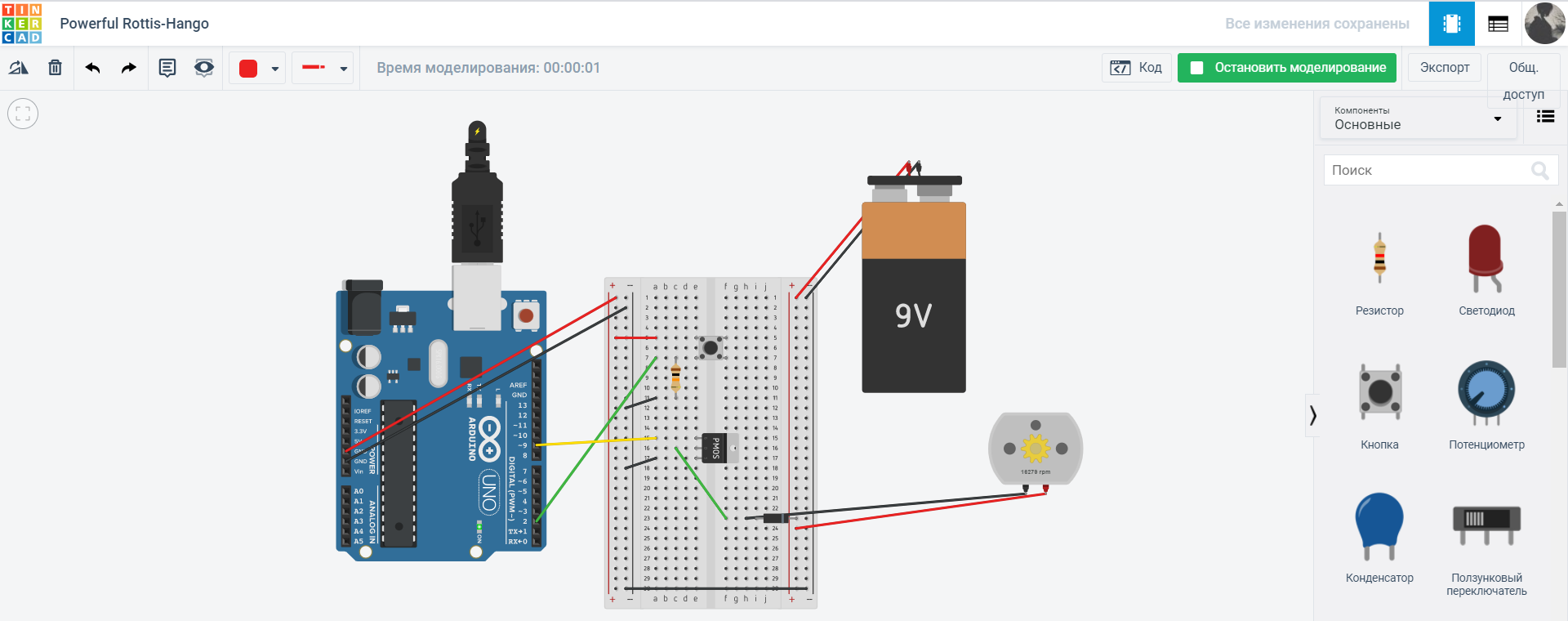
}

prevSwitchState = switchState;

}  
  
**6.** Provide link to the project.

<https://www.tinkercad.com/things/gkWi4uUgxD7>   
  
  
**Part 4. Project 9 from Arduino Projects Book.** **(25%)**  
*Build project 9 from Arduino Projects Book in Tinkercad.*

1. Provide screenshot of the circuit



**2.** Explain why we need MOSFETs?

**3.** Record voltage in the motor and current passed to the motor

|  |  |
| --- | --- |
| State | Value |
| Voltage in the motor (Switch ON) | 8.14 V |
| Current passed to the motor (Switch ON) | 5.50 A |
| Voltage in the motor (Switch OFF) | 0 V |
| Current passed to the motor (Switch OFF) | 0 A |

**4.** What is the difference between DC motor and Servo Motor?

Servomotors only rotate at a limited angle but are accurate and fast. And DC motors have continuous rotation and also fast.

1. In which projects you would use DC motors and in which projects you would prefer Servo Motors? (3 projects for each motor)

DC motors: car wheels, fans, mill. Servo motors: mechanical arm, Arduino sunflower, walking robot.

1. Provide codes with comments

const int switchPin = 2; //name constants and variables

const int motorPin = 9;

int switchState = 0;

void setup() { //declare the directions

pinMode(motorPin, OUTPUT);

pinMode(switchPin, INPUT);

}

void loop(){ //read input and give output

switchState = digitalRead(switchPin);

if (switchState == HIGH) {

digitalWrite(motorPin, HIGH);

}

else {

digitalWrite(motorPin, LOW);

}

**7.** Provide link to the project

<https://www.tinkercad.com/things/fc6dx1KJKiq>