



آزمایش یعقوب برقی

• هدف آزمایش

شبیه سازی یک vending machine با ساندویچ ها و نوشیدنی های گرم و سرد

• وسایل مورد نیاز

• برد آردوینو مگا

در این برد تعیین میکنیم که چه منویی روی LCD نمایش داده شود و پس از انتخاب محصول و تعداد آن به کمک کیپد، به موتور servo ی مورد نظر فرمان میدهم که به تعداد کالای درخواستی باز و بسته شود.

• موتور SERVO

به تعداد هر یک از محصولات یک موتور servo خواهیم داشت. (۱۲ عدد) این موتور در اصل نقش یک در را دارد که با هر بار باز و بسته شدن آن یک عدد خوراکی در اختیار ما قرار میگیرد.

• LCD

برای نمایش وضعیت فعلی در از آن استفاده میکنیم. خوراکی هایی که در دسته خوراکی های فعلی قابل انتخاب هستند قابل رویت بوده و میتوان مبلغ پرداختی را نیز روی آن مشاهده کرد.

• Keypad

کنترل این دستگاه توسط کیپد انجام میشود. در بخش اول میتوان به کمک جدول زیر با دستگاه ارتباط برقرار کرد:

عمل	کلید
Go right.	+
Go left.	-
Go to root.	ON/C
select	=

در ادامه که از ما تعداد کالا خواسته میشود نیز میتوان با استفاده از اعداد، تعداد درخواستی محصول را به میکرو اطلاع رسانی کنیم.

• کتابخانه MenuBackend

MenuBackend (menuUse, menuChange)

ساخت یک شی از MenuBackend به کمک این constructor، با آرگومان های menuChange و menuUse که از جنس تابع میباشد.



آزمایش یعقوب برقی

menuChange

تابعی که به هنگام تغییر وضعیت بین گزینه های منو به صورت خودکار فراخوانی میشود . پیاده سازی آن در قسمت کد بیان شده است.

menuUse

وقتی میخواهیم آیتم منو را مورد استفاده قرار دهیم آن را فراخوانی میکنیم که توسط خود ما پیاده سازی خواهد شد.

MenuItem

کلاسی که برای ساختن آیتم های منو از آن استفاده میشود.

getRoot()

بالاترین سطح منو را برمیگرداند.

toRoot()

یا این تابع میتواند به بالا ترین سطح منو رسید.

Add()

ساخت یک زیر شاخه برای یک آیتم منو.

AddRight()

افزودن یک sibling به آیتم منویی که در آن قرار داریم. این افزودن از سمت راست آیتم قبلی صورت خواهد گرفت.

getCurrent()

این تابع آیتم ای را که اکنون در آن هستیم برمیگرداند .

moveDown()

حرکت به سمت زیر شاخه پایینی

moveRight()

حرکت به سمت زیر شاخه راست

moveLeft()

حرکت به سمت زیر شاخه چپ

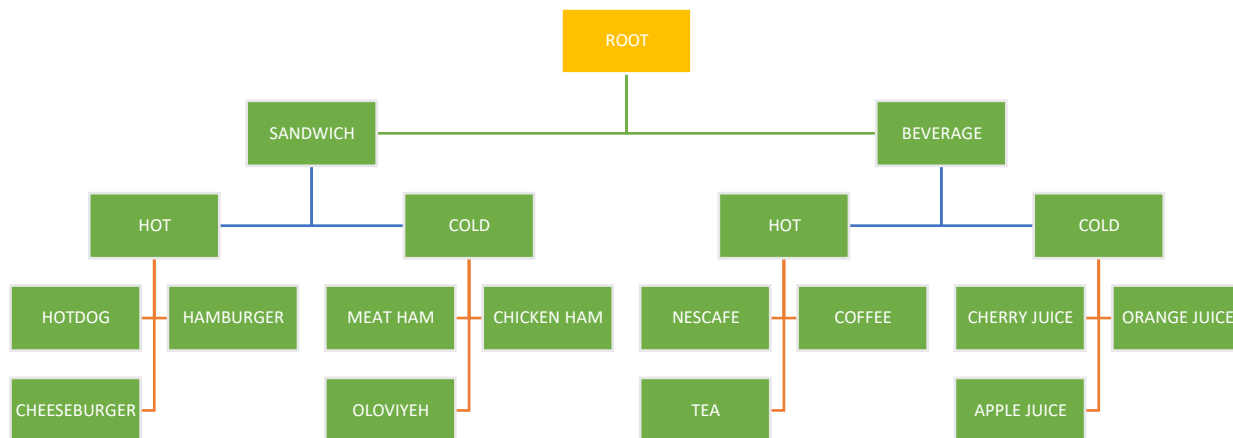
moveUp()

حرکت به سمت زیر شاخه بالا



آزمایش یعقوب برقی

• شرح آزمایش



برای ساختن منویی که درخت بالا را تولید کند ابتدا آیتم های زیر را تعریف میکنیم:

```
MenuItem sandwiches = MenuItem ("sandwiches");
    MenuItem hot_sandwiches = MenuItem ("hot sandwiches");
        MenuItem hotdog = MenuItem ("hotdog");
        MenuItem hamburger = MenuItem ("hamburger");
        MenuItem cheeseburger = MenuItem ("cheeseburger");
    MenuItem cold_sandwiches= MenuItem ("cold sandwiches");
        MenuItem meat_ham = MenuItem ("meat ham");
        MenuItem chicken_ham = MenuItem ("chicken ham");
        MenuItem oloviyeh = MenuItem ("oloviyeh");
MenuItem beverages = MenuItem ("beverages");
    MenuItem hot_beverages = MenuItem ("hot beverages");
        MenuItem nescafe = MenuItem ("nescafe");
        MenuItem coffee = MenuItem ("coffee");
        MenuItem tea = MenuItem ("tea");
    MenuItem cold_beverages= MenuItem ("cold beverages");
        MenuItem cherry_juice = MenuItem ("cherry juice");
```



آزمایش یعقوب برقی

```
MenuItem orange_juice = MenuItem ("orange juice");
MenuItem apple_juice = MenuItem ("apple juice");
```

سپس از کد زیر استفاده میکنیم:

```
//creating the menu
menu.getRoot().add(sandwiches);
sandwiches.addRight(beverages);
sandwiches.add(hot_sandwiches).addRight(cold_sandwiches);
hot_sandwiches.add(hotdog).addRight(hamburger).addRight(cheeseburger);
cold_sandwiches.add(meat_ham).addRight(chicken_ham).addRight(oloviyeh);
beverages.add(hot_beverages).addRight(cold_beverages);
hot_beverages.add(nescafe).addRight(coffee).addRight(tea);
cold_beverages.add(cherry_juice).addRight(orange_juice).addRight(apple_juice);
menu.toRoot();
```

در تابع menuChange با توجه به وضعیتی که در آن قرار گرفته ایم LCD را آپدیت میکنیم. علاوه بر آپدیت شدن LCD در گرهِ های برگ (که کالا هستند نه کتگوری غذا) دو متغیر selected_servo و selected_price نیز مقدار دهی میشوند.

```
void menu_change (MenuChangeEvent changed){ // after changing our current position
    MenuItem newMenuItem = changed.to;
    .
    .
    .
    else if ( newMenuItem.getName () == "nescafe" ){
        selected_servo = NESCAFE_servo;
        selected_price = NESCAFE;
        lcd.setCursor(0, 0);
        lcd.print("<hot beverages>");
        lcd.setCursor(0, 1);
        lcd.print("    nescafe    +");
    }
    .
    .
    .
}
```

در menuUse تعداد کالا از ما پرسیده میشود و به آن تعداد پس از محاسبه قیمت، موتور مورد نظر توسط تابع زیر به تعداد خواسته شده باز و بسته میشود. (که قیمت کالاها در یک آرایه ذخیره شده است.)

```
void menu_use (MenuUseEvent used){// after selecting the product...
    // update screen
    lcd.clear ();
    lcd.setCursor(0,0);
```



آزمایش یعقوب برقی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصومزاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
lcd.print (> ");
lcd.print (used.item.getName ());
lcd.setCursor(0, 1);
lcd.print (> How many? ");
// get the number of products
String number_string = "";
char n = kpd.waitForKey();
while (n != '='){
    number_string += n;
    lcd.print(n);
    n = kpd.waitForKey();
}
int number = number_string.toInt();
// get the products from Yaghoub:)
open_servo(selected_servo, number);
}
```

برای باز و بسته شدن موتور مورد نظر از تابع زیر استفاده میکنیم:

```
void open_servo(Servo servo, int n){
    //print
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print (n);
    lcd.print ("*");
    lcd.print (prices[13 - selected_price]);
    long result = n*prices[13 - selected_price];
    lcd.print ("=");
    lcd.setCursor(0, 1);
    lcd.print (result);
    lcd.print (" : ");
    lcd.print(n);
    //wait
    delay(DELAY_PERIOD);
    for (int i = 0; i < n; i++){
        //open
        servo.write(180);
        // update screen
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print (n);
        lcd.print ("*");
        lcd.print (prices[13 - selected_price]);
        lcd.print ("=");
        lcd.setCursor(0, 1);
        lcd.print (result);
    }
}
```



آزمایش یعقوب برقی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصوم زاده

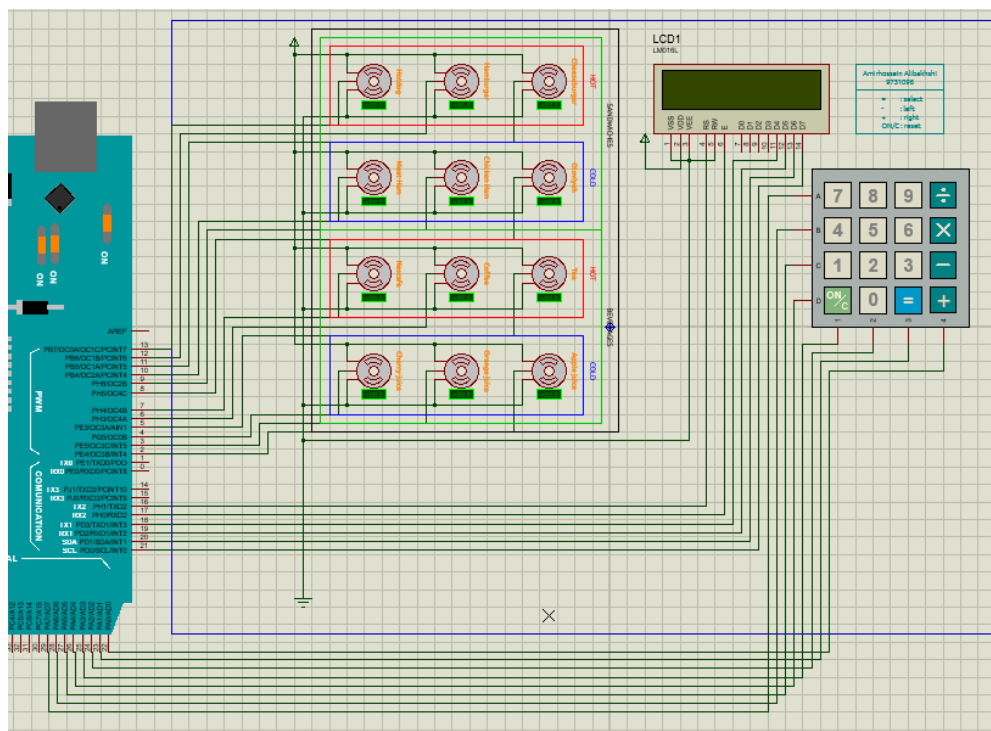
درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
lcd.print (" : ");
lcd.print(n-i-1);
//wait
delay(DELAY_PERIOD);
//close
servo.write(0);
//wait
delay(DELAY_PERIOD * 2);
}
//go to root
menu.toRoot ();
}
```

تابعی نیز وجود دارد که در هر بار اجرای تابع loop بعد از خواندن مقدار کیپد فراخوانی میشود. کار این تابع این است که با توجه به اینکه چه کلیدی فشار داده شده است، داخل منو حرکت کنیم:

```
void navigateMenus (char key){
MenuItem currentMenu = menu.getCurrent ();
switch ( key ){
case '=': //enter
    if ( !(currentMenu.moveDown ()) ){ // No child found; the node is a leaf.
        menu.use();
    }
    else{ // Go to the next level
        menu.moveDown();
    }
    break;
case 'R': // reset. Go back to the root.
    menu.toRoot ();
    break;
case '+': //right
    menu.moveRight ();
    break;
case '-': //left
    menu.moveLeft ();
    break;
// update the screen
lcd.clear();
lcd.setCursor(0, 0);
lcd.print(key);
delay(1000);
}
}
```

- مدار



• کد (کامل)

```

/*
Microprocessor and Assembly Language Lab
Fall 1399
Amirhossein Alibakhshi (9731096)
Lab #6 (Yaghoob Barghi)
*/

// importing libraries
#include <LiquidCrystal.h>
#include <Keypad.h>
#include <Servo.h>
#include <MenuBackend.h>

#define RS 16
#define EN 17
#define D4 18
#define D5 19
#define D6 20
#define D7 21
#define DELAY_PERIOD 500

```



آزمایش یعقوب برقی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصومزاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
#define BUZZER_PORT 10
#define SERVO_PORT 9
#define LED_PORT 11

#define HOTDOG 13
#define HAMBURGER 12
#define CHEESEBURGER 11
#define MEAT_HAM 10
#define CHICKEN_HAM 9
#define OLOVIYEH 8
#define NESCAFE 7
#define COFFEE 6
#define TEA 5
#define CHERRY_JUICE 4
#define ORANGE_JUICE 3
#define APPLE_JUICE 2

const byte ROWS = 4; // Four rows
const byte COLS = 4; //Four columns
// Define the Keymap
char keys[ROWS][COLS] = {
  {'7','8','9','/'},
  {'4','5','6','*'},
  {'1','2','3','-'},
  {'R','0','=','+'}
};

byte rowPins[ROWS] = { 29, 28, 27, 26 }; // Connect keypad ROW0, ROW1, ROW2 and ROW3 to these Arduino pins.
byte colPins[COLS] = { 25, 24, 23, 22 }; // Connect keypad COL0, COL1 and COL 2 to these Arduino pins.
Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

LiquidCrystal lcd(RS, EN, D4, D5, D6, D7);

Servo HOTDOG_servo;
Servo HAMBURGER_servo;
Servo CHEESEBURGER_servo;
Servo MEAT_HAM_servo;
Servo CHICKEN_HAM_servo;
Servo OLOVIYEH_servo;
Servo NESCAFE_servo;
Servo COFFEE_servo;
Servo TEA_servo;
Servo CHERRY_JUICE_servo;
```




آزمایش یعقوب برقی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصوم زاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
Servo ORANGE_JUICE_servo;
Servo APPLE_JUICE_servo;
Servo selected_servo;
int selected_price;
long prices[12] = {20000, // HOTDOG
                   25000, // HAMBURGER
                   30000, // CHEESEBURGER
                   15000, // MEAT_HAM
                   15000, // CHICKEN_HAM
                   13000, // OLOVIYEH
                   5000,  // NESCAFE
                   4000,  // COFFEE
                   3500,  // TEA
                   4000,  // CHERRY_JUICE
                   4000,  // ORANGE_JUICE
                   4000}; // APPLE_JUICE

// prototypes for menu functions
void menu_change (MenuChangeEvent changed);
void menu_use (MenuUseEvent changed);

MenuBackend menu = MenuBackend (menu_use, menu_change);

MenuItem sandwiches = MenuItem ("sandwiches");
    MenuItem hot_sandwiches = MenuItem ("hot sandwiches");
        MenuItem hotdog = MenuItem ("hotdog");
        MenuItem hamburger = MenuItem ("hamburger");
        MenuItem cheeseburger = MenuItem ("cheeseburger");
    MenuItem cold_sandwiches= MenuItem ("cold sandwiches");
        MenuItem meat_ham = MenuItem ("meat ham");
        MenuItem chicken_ham = MenuItem ("chicken ham");
        MenuItem oloviyeh = MenuItem ("oloviyeh");
MenuItem beverages = MenuItem ("beverages");
    MenuItem hot_beverages = MenuItem ("hot beverages");
        MenuItem nescafe = MenuItem ("nescafe");
        MenuItem coffee = MenuItem ("coffee");
        MenuItem tea = MenuItem ("tea");
    MenuItem cold_beverages= MenuItem ("cold beverages");
        MenuItem cherry_juice = MenuItem ("cherry juice");
        MenuItem orange_juice = MenuItem ("orange juice");
        MenuItem apple_juice = MenuItem ("apple juice");

void setup() {
    //set servo pins
    pinMode(HOTDOG, OUTPUT);
```



آزمایش یعقوب برقی

```
pinMode(HAMBURGER, OUTPUT);
pinMode(CHEESEBURGER, OUTPUT);
pinMode(MEAT_HAM, OUTPUT);
pinMode(CHICKEN_HAM, OUTPUT);
pinMode(OLOVIYEH, OUTPUT);
pinMode(NESCAFE, OUTPUT);
pinMode(COFFEE, OUTPUT);
pinMode(TEA, OUTPUT);
pinMode(CHERRY_JUICE, OUTPUT);
pinMode(ORANGE_JUICE, OUTPUT);
pinMode(APPLE_JUICE, OUTPUT);

//attach servo pins
HOTDOG_servo.attach(HOTDOG , 1000 , 2000);
HAMBURGER_servo.attach(HAMBURGER , 1000 , 2000);
CHEESEBURGER_servo.attach(CHEESEBURGER , 1000 , 2000);
MEAT_HAM_servo.attach(MEAT_HAM , 1000 , 2000);
CHICKEN_HAM_servo.attach(CHICKEN_HAM , 1000 , 2000);
OLOVIYEH_servo.attach(OLOVIYEH , 1000 , 2000);
NESCAFE_servo.attach(NESCAFE , 1000 , 2000);
COFFEE_servo.attach(COFFEE , 1000 , 2000);
TEA_servo.attach(TEA , 1000 , 2000);
CHERRY_JUICE_servo.attach(CHERRY_JUICE , 1000 , 2000);
ORANGE_JUICE_servo.attach(ORANGE_JUICE , 1000 , 2000);
APPLE_JUICE_servo.attach(APPLE_JUICE , 1000 , 2000);

//initialize the servos
HOTDOG_servo.write(0);
HAMBURGER_servo.write(0);
CHEESEBURGER_servo.write(0);
MEAT_HAM_servo.write(0);
CHICKEN_HAM_servo.write(0);
OLOVIYEH_servo.write(0);
NESCAFE_servo.write(0);
COFFEE_servo.write(0);
TEA_servo.write(0);
CHERRY_JUICE_servo.write(0);
ORANGE_JUICE_servo.write(0);
APPLE_JUICE_servo.write(0);

//creating the menu
menu.getRoot().add(sandwiches);
sandwiches.addRight(beverages);
sandwiches.add(hot_sandwiches).addRight(cold_sandwiches);
hot_sandwiches.add(hotdog).addRight(hamburger).addRight(cheeseburger);
```



آزمایش یعقوب برقی

```
cold_sandwiches.add(meat_ham).addRight(chicken_ham).addRight(oloviyeh);
beverages.add(hot_beverages).addRight(cold_beverages);
hot_beverages.add(nescafe).addRight(coffee).addRight(tea);
cold_beverages.add(cherry_juice).addRight(orange_juice).addRight(apple_juice);
menu.toRoot();

lcd.begin(16, 2);
lcd.print("  A.Alibakhshi  ");
lcd.setCursor(0, 1);
lcd.print(" YAGHOOB-BARGHI ");
delay(DELAY_PERIOD * 2);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("< Main Menu >");
lcd.setCursor(0, 1);
lcd.print("Press = to start");
}

void loop() {
  char key = kpd.waitForKey();
  navigateMenus(key);
}

void menu_change (MenuChangeEvent changed){ // after changing our current position in the menu...
  MenuItem newItem = changed.to;
  if ( newItem.getName () == menu.getRoot () ){
    lcd.setCursor(0, 0);
    lcd.print("< Main Menu >");
    lcd.setCursor(0, 1);
    lcd.print("Press = to start");
  }
  else if ( newItem.getName () == "sandwiches" ){
    lcd.setCursor(0, 0);
    lcd.print("<      Menu      >");
    lcd.setCursor(0, 1);
    lcd.print("  sandwiches  +");
  }
  else if ( newItem.getName () == "hot sandwiches" ){
    lcd.setCursor(0, 0);
    lcd.print("< sandwiches >");
    lcd.setCursor(0, 1);
    lcd.print(" hot sandwiches+");
  }
}
```



آزمایش یعقوب برقی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصوم زاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
}
else if ( newItem.getName () == "hotdog" ){
    selected_servo = HOTDOG_servo;
    selected_price = HOTDOG;
    lcd.setCursor(0, 0);
    lcd.print ("<hot sandwiches>");
    lcd.setCursor(0, 1);
    lcd.print ("    hotdog    +");
}
else if ( newItem.getName () == "hamburger" ){
    selected_servo = HAMBURGER_servo;
    selected_price = HAMBURGER;
    lcd.setCursor(0, 0);
    lcd.print ("<hot sandwiches>");
    lcd.setCursor(0, 1);
    lcd.print ("-  hamburger  +");
}
else if ( newItem.getName () == "cheeseburger" ){
    selected_servo = CHEESEBURGER_servo;
    selected_price = CHEESEBURGER;
    lcd.setCursor(0, 0);
    lcd.print ("<hot sandwiches>");
    lcd.setCursor(0, 1);
    lcd.print ("- cheeseburger  ");
}
else if ( newItem.getName () == "cold sandwiches" ){
    lcd.setCursor(0, 0);
    lcd.print ("<  sandwiches  >");
    lcd.setCursor(0, 1);
    lcd.print ("-coldSandwiches ");
}
else if ( newItem.getName () == "meat ham" ){
    selected_servo = MEAT_HAM_servo;
    selected_price = MEAT_HAM;
    lcd.setCursor(0, 0);
    lcd.print ("<coldSandwiches>");
    lcd.setCursor(0, 1);
    lcd.print ("    meat ham    +");
}
else if ( newItem.getName () == "chicken ham" ){
    selected_servo = CHICKEN_HAM_servo;
    selected_price = CHICKEN_HAM;
    lcd.setCursor(0, 0);
    lcd.print ("<coldSandwiches>");
    lcd.setCursor(0, 1);
```



آزمایش یعقوب برقی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصوم زاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
    lcd.print ("- chicken ham +");
}
else if ( newItem.getName () == "oloviyeh" ){
    selected_servo = OLOVIYEH_servo;
    selected_price = OLOVIYEH;
    lcd.setCursor(0, 0);
    lcd.print ("<coldSandwiches>");
    lcd.setCursor(0, 1);
    lcd.print ("-   oloviyeh   ");
}
else if ( newItem.getName () == "beverages" ){
    lcd.setCursor(0, 0);
    lcd.print ("<   Menu   >");
    lcd.setCursor(0, 1);
    lcd.print ("-   beverages   ");
}
else if ( newItem.getName () == "hot beverages" ){
    lcd.setCursor(0, 0);
    lcd.print ("<   beverages >");
    lcd.setCursor(0, 1);
    lcd.print (" hot beverages +");
}
else if ( newItem.getName () == "nescafe" ){
    selected_servo = NESCAFE_servo;
    selected_price = NESCAFE;
    lcd.setCursor(0, 0);
    lcd.print ("<hot   beverages>");
    lcd.setCursor(0, 1);
    lcd.print ("   nescafe   +");
}
else if ( newItem.getName () == "coffee" ){
    selected_servo = COFFEE_servo;
    selected_price = COFFEE;
    lcd.setCursor(0, 0);
    lcd.print ("<hot   beverages>");
    lcd.setCursor(0, 1);
    lcd.print ("-   coffee   +");
}
else if ( newItem.getName () == "tea" ){
    selected_servo = TEA_servo;
    selected_price = TEA;
    lcd.setCursor(0, 0);
    lcd.print ("<hot   beverages>");
    lcd.setCursor(0, 1);
    lcd.print ("-   tea       ");
}
```



آزمایش یعقوب برقی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصوم زاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
}
else if ( newItem.getName () == "cold beverages" ){
    lcd.setCursor(0, 0);
    lcd.print ("< beverages >");
    lcd.setCursor(0, 1);
    lcd.print ("- cold beverages");
}
else if ( newItem.getName () == "cherry juice" ){
    selected_servo = CHERRY_JUICE_servo;
    selected_price = CHERRY_JUICE;
    lcd.setCursor(0, 0);
    lcd.print ("<cold beverages>");
    lcd.setCursor(0, 1);
    lcd.print (" cherry juice +");
}
else if ( newItem.getName () == "orange juice" ){
    selected_servo = ORANGE_JUICE_servo;
    selected_price = ORANGE_JUICE;
    lcd.setCursor(0, 0);
    lcd.print ("<cold beverages>");
    lcd.setCursor(0, 1);
    lcd.print ("- orange juice +");
}
else if ( newItem.getName () == "apple juice" ){
    selected_servo = APPLE_JUICE_servo;
    selected_price = APPLE_JUICE;
    lcd.setCursor(0, 0);
    lcd.print ("<cold beverages>");
    lcd.setCursor(0, 1);
    lcd.print ("- apple juice ");
}
}

void menu_use (MenuUseEvent used){// after selecting the priduct...
    // update screen
    lcd.clear ();
    lcd.setCursor(0,0);
    lcd.print ("> ");
    lcd.print (used.item.getName ());
    lcd.setCursor(0, 1);
    lcd.print ("> How many? ");
    // get the number of products
    String number_string = "";
    char n = kpd.waitForKey();
    while (n != '='){
```



آزمایش یعقوب برقی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصوم زاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
        number_string += n;
        lcd.print(n);
        n = kpd.waitForKey();
    }
    int number = number_string.toInt();
    // get the products from Yaghoub:)
    open_servo(selected_servo, number);
}

void open_servo(Servo servo, int n){
    //print
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print (n);
    lcd.print ("*");
    lcd.print (prices[13 - selected_price]);
    long result = n*prices[13 - selected_price];
    lcd.print ("=");
    lcd.setCursor(0, 1);
    lcd.print (result);
    lcd.print (" : ");
    lcd.print(n);
    //wait
    delay(DELAY_PERIOD);
    for (int i = 0; i < n; i++){
        //open
        servo.write(180);
        // update screen
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print (n);
        lcd.print ("*");
        lcd.print (prices[13 - selected_price]);
        lcd.print ("=");
        lcd.setCursor(0, 1);
        lcd.print (result);
        lcd.print (" : ");
        lcd.print(n-i-1);
        //wait
        delay(DELAY_PERIOD);
        //close
        servo.write(0);
        //wait
        delay(DELAY_PERIOD * 2);
    }
}
```



آزمایش یعقوب برقی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصوم زاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
//go to root
menu.toRoot ();
}

void navigateMenus (char key){
    MenuItem currentMenu = menu.getCurrent ();
    switch ( key ){
        case '=': //enter
            if ( !(currentMenu.moveDown ()) ){ // No child found; the node is a leaf.
                menu.use();
            }
            else{ // Go to the next Level
                menu.moveDown();
            }
            break;
        case 'R': // reset. Go back to the root.
            menu.toRoot ();
            break;
        case '+': //right
            menu.moveRight ();
            break;
        case '-': //left
            menu.moveLeft ();
            break;
        // update the screen
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(key);
        delay(1000);
    }
}
```