



Facultatea de Inginerie Electrică



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA



EUROPEAN UNIVERSITY
OF TECHNOLOGY

PCLP 2

Programarea calculatoarelor si limbaje de programare 2

PCLP2

An I semestrul II



"Coding is easy when you C it in action."

Cap. 13

Aplicatii C/C++ in inginerie electrica

13.1. Aplicatii cu circuite electrice

13.2. Aplicatii cu Arduino

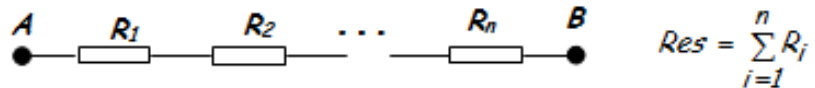
13.1. Aplicații cu circuite electrice

EXEMPLE

Rezistența echivalentă

Ex.1 Programul permite citirea de la tastatură a numărului de rezistențe, a valorilor lor în Ohmi și a tipului de conectare serie sau paralel și apoi calculează și afișează rezistența echivalentă corespunzătoare conectării a) serie și b) paralel (Fig. 12.1).

a) serie



b) paralel

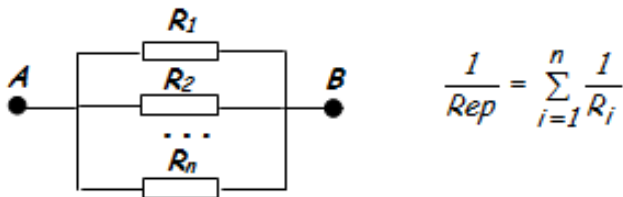


Fig.12.1. Rezistența echivalentă a conexiunii
a) serie, b) paralel

```
#include <iostream>
#include <vector>
using namespace std;
int main() {
    int n; char c; double Res = 0, Rep = 0;
    cout << "Programul calculeaza rezistența echivalentă\n";
    cout << "Introduceți nr de rezistențe: "; cin >> n;
    vector<double> R(n);
    cout << "Introduceți valorile rezistențelor în Ohmi\n";
    for (int i = 0; i < n; i++) {
        cout << "R[" << i << "]="; cin >> R[i]; }
    cout << "Cum sunt conectate rezistențele?\n s=serie p=paralel: "; cin >> c;
    // Calculul rezistenței echivalente pentru conexiunea în serie
    if (c == 's') {
        for (int i = 0; i < n; i++) Res += R[i];
        cout << "Rezistența echivalentă serie: R=" << Res << "\n"; }
    // Calculul rezistenței echivalente pentru conexiunea în paralel
    if (c == 'p') {
        for (int i = 0; i < n; i++) Rep += 1 / R[i];
        cout << "Rezistența echivalentă paralel: R=" << 1 / Rep << "\n"; }
    return 0;}

```

13.1. Aplicatii cu circuite electrice

Rezistenta echivalenta

Rezistente conectate in serie:

```
Programul calculeaza rezistenta echivalenta
Introduceti nr de rezistente
3
Introduceti valorile rezistentelor en Ohmi
R[0]=10
R[1]=20
R[2]=30
cum sunt conectate rezistentele?
s=serie p=paralel: s
Rezistenta echivalenta serie: R=60.000000
```

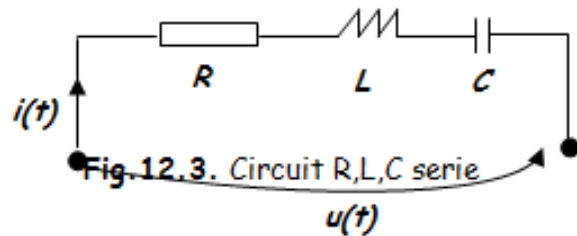
Rezistente conectate in paralel:

```
Programul calculeaza rezistenta echivalenta
Introduceti nr de rezistente
3
Introduceti valorile rezistentelor en Ohmi
R[0]=10
R[1]=20
R[2]=30
cum sunt conectate rezistentele?
s=serie p=paralel: p
Rezistenta echivalenta paralel: R=5.454545
```

13.1. Aplicații cu circuite electrice

Rezistența echivalentă

Ex.2 Într-un circuit R,L,C serie (Fig.12.3) aflat în regim armonic se cunosc valorile tuturor parametrilor: $R=10\ \Omega$, $L=10\text{mH}$, $C=5\mu\text{F}$, $U=220\ \text{V}$, $f=50\ \text{Hz}$. Să se scrie un program care să determine valoarea instantanee a curentului din circuit.



Rezolvare:

Pe baza mărimilor cunoscute se calculează:

- Pulsăția: $\omega=2\pi f=314\ \text{rad/s}$.

- Impedanța circuitului dat: $Z = \sqrt{R^2 + \left(\omega L - \frac{1}{\omega C}\right)^2}$ (Ω)

- Valoarea efectivă a curentului: $I = \frac{U}{Z}$ (A)

- Defazajul: $\varphi = \arctg \frac{\omega L - \frac{1}{\omega C}}{R}$ (radiani)

- Valoarea instantanee a curentului: $i = I \cdot \sqrt{2} \sin(\omega t - \varphi)$

EXEMPLE

```
#include <iostream>
#include <cmath>
#define PI 3.14159265358979
using namespace std;
int main() {
double R, L, C, U, I, f, w, Z, phi;
cout << "R[ohm]="; cin >> R;
cout << "L[H]="; cin >> L;
cout << "C[F]="; cin >> C;
cout << "U[V]="; cin >> U;
cout << "f[Hz]="; cin >> f;
w = 2 * PI * f;
Z = sqrt(pow(R, 2) + pow((w * L - 1 / (w * C)), 2));
phi = atan((w * L - 1 / (w * C)) / R);
I = U / Z;
cout << "Z[ohm]=" << Z << endl;
cout << "I[A]=" << I << "\t phi[degrees]=" << phi * 180 / PI ;
return 0;}
```

```
R[ohmi]=10
L[H]=0.01
C[F]=0.000005
U[V]=220
f[Hz]=50
Z[ohmi]=633.557104
I[A]=0.347246    phi[grade]=-89.095612
```

13.1. Aplicații cu circuite electrice

Divizor de tensiune

Ex.5 Să se scrie un program care să determine valoarea efectivă complexă a tensiunii \underline{U}_2 din circuitul divizor de tensiune (Fig.12.7) care funcționează în regim permanent de c.a. pentru care se cunosc valorile celor 2 impedanțe \underline{Z}_1 și \underline{Z}_2 precum și valoarea efectivă a tensiunii $U=220$ V.

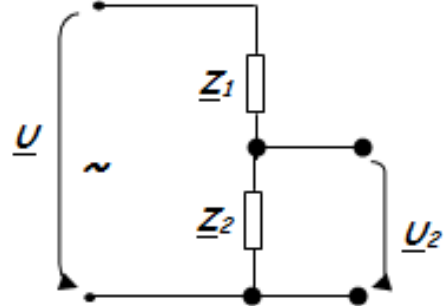
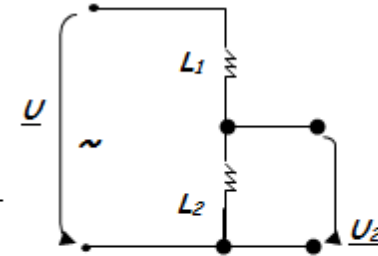


Fig.12.7. Divizor de tensiune

Rezolvare: sunt 3 cazuri particulare:

a) divizor inductiv de tensiune

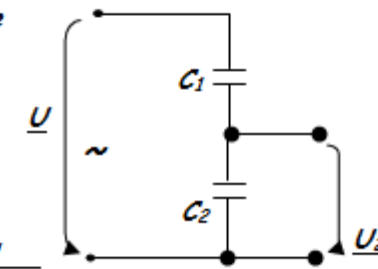
$$\begin{cases} \underline{Z}_1 = j\omega L_1 \\ \underline{Z}_2 = j\omega L_2 \end{cases}$$
$$\underline{U}_2 = \underline{U} \frac{j\omega L_2}{j\omega L_1 + j\omega L_2} = \underline{U} \frac{L_2}{L_1 + L_2}$$



unde $U=220$ V (U origine de fază). Caz particular: $L_1=10$ mH, $L_2=20$ mH

b) divizor capacitiv de tensiune

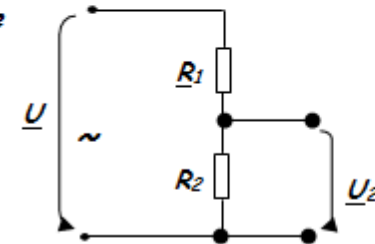
$$\begin{cases} \underline{Z}_1 = \frac{1}{j\omega C_1} \\ \underline{Z}_2 = \frac{1}{j\omega C_2} \end{cases}$$
$$\underline{U}_2 = \underline{U} \frac{\frac{1}{j\omega C_2}}{\frac{1}{j\omega C_1} + \frac{1}{j\omega C_2}} = \underline{U} \frac{C_1}{C_1 + C_2}$$



unde $U=220$ V (U origine de fază). Caz particular: $C_1=10$ μ F, $C_2=20$ μ F.

c) Divizor rezistiv de tensiune

$$\begin{cases} \underline{Z}_1 = R_1 \\ \underline{Z}_2 = R_2 \end{cases}$$
$$\underline{U}_2 = \underline{U} \frac{R_2}{R_1 + R_2}$$



unde $U=220$ V (U origine de fază). Caz particular: $R_1=10$ Ω , $R_2=20$ Ω .

13.1. Aplicatii cu circuite electrice

Divizor de tensiune

EXEMPLE

```
#include <stdio.h>
#include <conio.h>
int main()
{char ch; double L1,L2,R1,R2,C1,C2, U;
printf("\nAlegeti tipul de divizor de tensiune:");
printf("\nInductiv=I, Capacitiv=C,Rezistiv=R :"); scanf("%c", &ch);
if (ch=='I' || ch=='i')
{printf("\nL1[H]=");scanf("%lf", &L1); printf("L2[H]=");scanf("%lf", &L2);
printf("U[V]=");scanf("%lf", &U); printf("U2[V]=%lf", U*L2/(L1+L2));}
else if (ch=='C' || ch=='c')
{printf("\nC1[F]=");scanf("%lf", &C1); printf("C2[F]=");scanf("%lf", &C2);
printf("U[V]=");scanf("%lf", &U); printf("U2[V]=%lf", U*C1/(C1+C2));}
else if (ch=='R' || ch=='r')
{printf("\nR1[ohmi]=");scanf("%lf", &R1);
printf("R2[ohmi]=");scanf("%lf", &R2);
printf("U[V]=");scanf("%lf", &U);
printf("U2[V]=%lf", U*R2/(R1+R2));} return 0;}
```

```
Alegeti tipul de divizor de tensiune:
Inductiv=I, Capacitiv=C,Rezistiv=R :r
```

```
R1[ohmi]=2
R2[ohmi]=3
U[V]=220
U2[V]=132.000000
```

```
Alegeti tipul de divizor de tensiune:
Inductiv=I, Capacitiv=C,Rezistiv=R :I
```

```
L1[H]=10
L2[H]=20
U[V]=220
U2[V]=146.666667
```

```
Alegeti tipul de divizor de tensiune:
Inductiv=I, Capacitiv=C,Rezistiv=R :C
```

```
C1[F]=2
C2[F]=3
U[V]=220
U2[V]=88.000000
```

13.1. Aplicații cu circuite electrice

Circuite electrice

Ex.3 Se consideră circuitul de curent continuu (c.c.) din Fig. 12.5, în care se cunosc valorile rezistențelor și tensiunilor electromotoare: $R_1=R_2=R_3=10\ \Omega$ și $E_1=E_2=100\ V$, și se cere să se scrie programul care să determine și afișeze valorile curenților: I_1, I_2, I_3 utilizând teoremele lui Kirchhoff.

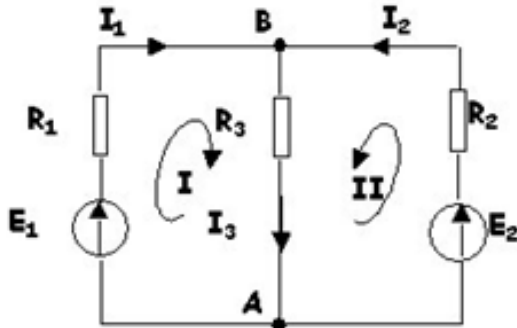


Fig.12.5. Circuitul 1 de c.c.

Rezolvare: Problema se reduce la rezolvarea unui sistem de ecuații algebrice liniare prin metoda indirectă, ecuații care rezultă din aplicarea teoremelor lui Kirchhoff, astfel:

$$\begin{cases} I_1 R_1 + I_3 R_3 = E_1 & \text{Teorema II a lui Kirchhoff -Bucla I} \\ I_2 R_2 + I_3 R_3 = E_2 & \text{Teorema II a lui Kirchhoff -Bucla II} \\ I_1 + I_2 - I_3 = 0 & \text{Teorema I a lui Kirchhoff -Nod A} \end{cases}$$

Forma matricială a sistemului de ecuații algebrice liniare este:

$$\begin{bmatrix} R_1 & 0 & R_3 \\ 0 & R_2 & R_3 \\ 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ 0 \end{bmatrix} \Leftrightarrow \mathbf{R} \mathbf{I} = \mathbf{E} \Rightarrow \mathbf{I} = \mathbf{R}^{-1} \mathbf{E}$$

unde matricile \mathbf{E} și \mathbf{R} se cunosc, iar matricea curenților \mathbf{I} trebuie calculată.

Înlocuind valorile cunoscute $R_1=R_2=R_3=10\ \Omega$ și $E_1=E_2=100\ V$ în relația matricială se obține:

$$\begin{bmatrix} 10 & 0 & 10 \\ 0 & 10 & 10 \\ 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} 100 \\ 100 \\ 0 \end{bmatrix} \Rightarrow \mathbf{I} = \mathbf{R}^{-1} \mathbf{E}$$

$$\mathbf{R}^{-1} = \begin{bmatrix} 0.067 & -0.033 & 0.333 \\ -0.033 & 0.067 & 0.333 \\ 0.033 & 0.033 & -0.333 \end{bmatrix}, \mathbf{I} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} 3.4 \\ 3.4 \\ 6.6 \end{bmatrix}$$

13.1. Aplicatii cu circuite electrice

Circuite electrice

```
//Programul ex3.c
#include <stdio.h>
#define L 10
int a1,b1;
//L=dimensiunea maxima a matricilor
//functia transp determina matricea transpusa (Mtransp) a matricii M
void transp(double M[L][L], double Mtransp[L][L])
{int i, j;
for(i = 0; i < L; i++)
    for(j = 0; j < L; j++)    Mtransp[j][i] = M[i][j];
}
//functia creerm determina suprimarea liniei l1 și a coloanei c1 din //matricea A și creeaza
matricea B în care se pastreaza matricea ramasa //aceasta functie este utila la calculul
determinantului
void creerm(double A[L][L], double B[L][L], int l1, int c1)
{int a, b, c = 0, d;
for(a = 0; a < L; a++)
{d = 0;
if(a != l1)
    {for(b = 0; b < L; b++)
    if(b != c1) {B[c][d] = A[a][b]; d++; }
    c++;
    }}}
//functia exposant determina daca un nr este multiplu de 2 și returneaza
//valoarea 1 daca este adevarat și -1 în caz contrar functia este folosita pentru calculul
//determinantului
double exposant(int nr)
{if (!(nr % 2))    return (1);
else return (-1);
}
//functia determinanta calculeaza determinantul unei matrici
```

```
//functia determinanta calculeaza determinantul unei matrici
double determinanta(double Matrice[L][L], int l1)
{int a;
double Temp[L][L], val = 0;
if(l1 == 1) return (Matrice[0][0]);
for(a = 0; a < L; a++)
    {creerm(Matrice, Temp, a, 0);
    val += (exposant(a) * Matrice[a][0] * determinanta(Temp, (l1 - 1)));
    }
}
//functia multiply inmulteste 2 matrici A*B și returneaza rezultatul în //matricea C
void multiply(double A[L][L], double B[L][L], double C[L][L])
{int a, b, c;
for(a = 0; a < L; a++)
    for(b = 0; b < L; b++)
        C[a][b] = 0;
for(a = 0; a < L; a++)
    for(b = 0; b < L; b++)
        for(c = 0; c < L; c++)    C[a][b] += A[a][c] * B[c][b];
}
//functia multiplyr inmulteste matricea A cu un nr. real matricea rezultat
//este B
void multiplyr(double x, double A[L][L], double B[L][L])
{int a, b;
for(a = 0; a < L; a++)
    for(b = 0; b < L; b++)    B[a][b] = A[a][b] * x;
}
}
```

13.1. Aplicatii cu circuite electrice

Circuite electrice

```
//functia mplusm realizeaza operatia C=A-B
void mplusm(double A[L][L], double B[L][L], double C[L][L])
{int i, j;
for(i = 0; i < L; i++)
    for(j = 0; j < L; j++)    C[i][j] = A[i][j] + B[i][j];
}
//functia scadere realizeaza operatia C=A-B
void scadere(double A[L][L], double B[L][L], double C[L][L])
{int i, j;
for(i = 0; i < L; i++)
    for(j = 0; j < L; j++)    C[i][j] = A[i][j] - B[i][j];
}
//functia initializ initializeaza matricea Matrice cu 0
void initializ(double Matrice[L][L])
{int a, b;
for(a = 0; a < L; a++)
    for(b = 0; b < L; b++)    Matrice[a][b] = 0;
}
```

```
//functia a fis afiseaza matricea Matrice
void a fis(double Matrice[L][L], int l, int c)
{int a, b;
printf("\nIata matricea:");
printf("\n*****\n");
for(a = 0; a < l; a++)
    { for(b = 0; b < c; b++)
        printf("%7.3lf ", Matrice[a][b]);    printf("\n");
    }
printf("*****\n");
printf("Apasati pe ENTER pentru a continua...");
fflush(stdin);
getchar();
}
//functia citire realizeaza citirea unei matrici
void citire(double Matrice[L][L], int l, int c)
{int i, j;
initializ(Matrice);
printf("\nCate linii are matricea (MAX: %d)? ", L);
do
    { fflush(stdin);    scanf("%d",&l); a1=l;
    } while ((l < 1) || (l > L));
printf("Cate coloane are matricea (MAX: %d)? ", L);
do
    { fflush(stdin);    scanf("%d",&c); b1=c;
    } while ((c < 1) || (c > L));
for(i = 0; i < l; i++)
    for(j = 0; j < c; j++)
        {printf("A[%d][%d]= ", i + 1, j + 1);fflush(stdin);
        scanf("%lf", &Matrice[i][j]);}
a fis (Matrice, l, c);
}
```

```
Apasati pe ENTER pentru a continua...
Bine ati venit in acest meniu !
-----
0) Exit
1) Introduceti matricea LxC (A)
2) Introduceti matricea LxC (B)
3) Introduceti un nr. real:
11) Calculeaza transpusa lui A
13) Calculeaza determinantul lui A
15) Calculeaza inversa lui A
-----
Alegeti optiunea dorita: 15
Multumesc.
-----
Matricea inversa este:

Iata matricea:
*****
    0.067   -0.033    0.333
   -0.033    0.067    0.333
    0.033    0.033   -0.333
*****
Apasati pe ENTER pentru a continua...
```

```
Bine ati venit in acest meniu !
-----
0) Exit
1) Introduceti matricea LxC (A)
2) Introduceti matricea LxC (B)
3) Introduceti un nr. real:
4) Calculeaza A + B
5) Calculeaza A - B
6) Calculeaza B - A
7) Calculeaza A * B
8) Calculeaza B * A
11) Calculeaza transpusa lui A
12) Calculeaza transpusa lui B
13) Calculeaza determinantul lui A
14) Calculeaza determinantul lui B
15) Calculeaza inversa lui A
16) Calculeaza inversa lui B
-----
Alegeti optiunea dorita: 7
Multumesc.
-----
Iata matricea:
*****
    3.400
    3.400
    6.600
*****
```

13.1. Aplicatii cu circuite electrice

Circuite electrice

Ex.: Realizati programul C++ care rezova problema:

O baterie cu t.e.m. $E = 24V$ și rezistența interioară $r = 1\Omega$ care este legată în circuitul unui bec cu rezistența $R = 71\Omega$.

Determionati si afisati:

- tensiunea electrică la bornele becului.
- intensitatea curentului de scurtcircuit

EXEMPLE

```
#include <iostream>
using namespace std;
int main(){float I,E,r,R,U,Isc;
cout<<"E=";cin >>E;
cout<<"r=";cin >>r;
cout<<"R=";cin >>R;
I=E/(R+r); cout <<"I="<<I<<endl;
U=I*R;cout <<"U="<<U<<endl;
Isc=E/r; cout<<"Isc="<<Isc<<endl;
return 0;}
```

```
E=24
r=1
R=71
I=0.333333
U=23.6667
Isc=24
```

13.1. Aplicații cu circuite electrice

Circuite electrice

Ex.: Realizați programul C++ care rezolvă problema:

Un prăjitor de pâine ce funcționează la tensiunea $U = 230\text{V}$ este străbătut de curentul de intensitate $I = 4\text{A}$, timpul cât trece curentul electric prin prăjitor este de 15min. Aflați:

- puterea disipată pe prăjitor
- rezistența prăjitorului
- energia disipată în 15 min.

EXEMPLE

```
#include <iostream>
using namespace std;
int main(){float U,I,P,R,W,t,tsec;
cout<<"U="; cin>>U;
cout<<"I=";cin >>I;
cout<<"t=";cin >>t;
tsec=t*60;cout<<"tsec="<<tsec<<endl;
P=U*I; cout <<"P="<<P<<endl;;
R=U/I; cout<<"R="<<R<<endl;
W=U*I*tsec; cout<<"W="<<W<<endl;
return 0;}
```

```
U=230
I=4
t=15
tsec=900
P=920
R=57.5
W=828000
```

13.2. Aplicatii cu Arduino

Tutoriale Arduino

Tutorials



GETTING STARTED

Are you brand new to Arduino? Just put your hands on a new Arduino product? Worry no more. Here you can find our kickstart guides: step-by-step instructions to get you started with Arduino right away, so you can focus your time on creating incredible things.



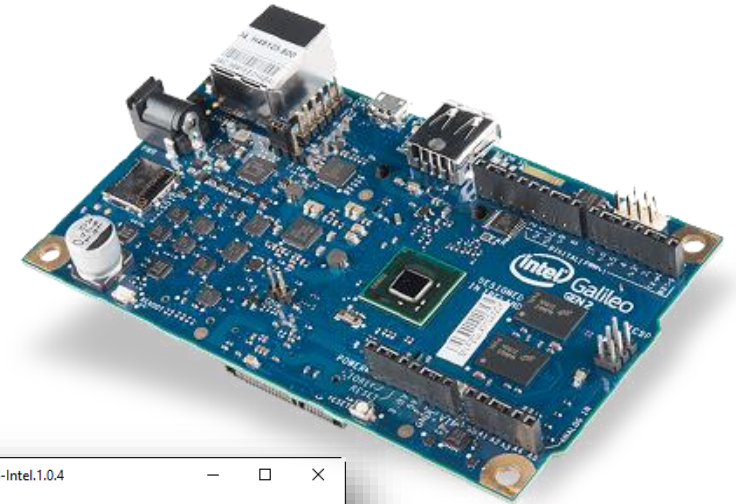
BUILT-IN EXAMPLES

Built-in Examples are sketches included in the Arduino Software (IDE); to open them, click on the toolbar menu: File > Examples. These simple programs demonstrate all basic Arduino commands. They span from a Bare Minimum Sketch to Digital and Analog IO, to the use of Sensors and Displays.



EXAMPLES FROM LIBRARIES

The Arduino Software (IDE) can be extended through the use of libraries, just like most programming platforms, to provide extra functionality to your sketches. These tutorials walk you through the Examples of a number of libraries that come installed with the IDE. To open them, click on the toolbar menu: File > Examples.

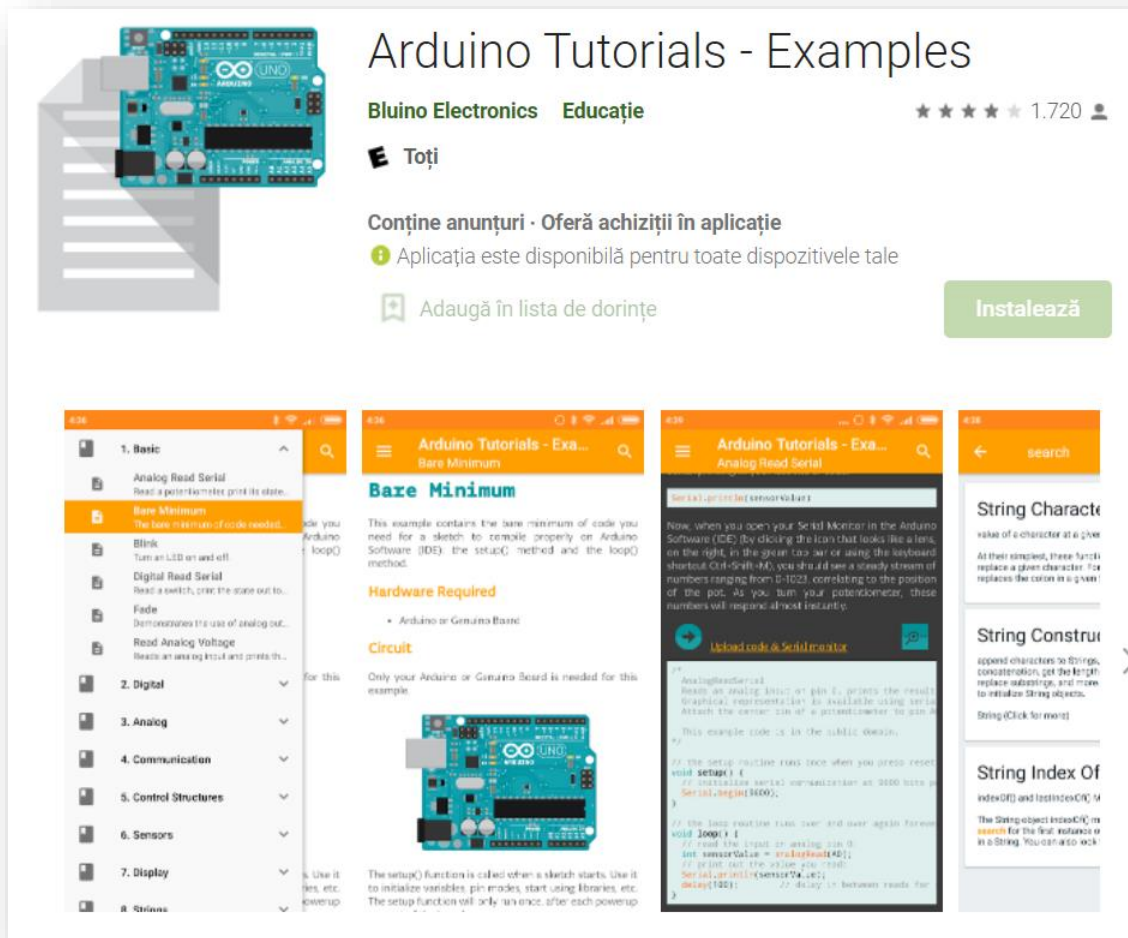


```
sketch_mar29a | Arduino 1.5.3-Intel.1.0.4
File Edit Sketch Tools Help
sketch_mar29a
void setup() {
  // put your setup code here, to run once:
}
void loop() {
  // put your main code here, to run repeatedly:
}
Intel® Galileo Gen2 on COM3
```

<https://www.arduino.cc/en/Tutorial/HomePage>

13.2. Aplicații cu Arduino

Tutoriale Arduino



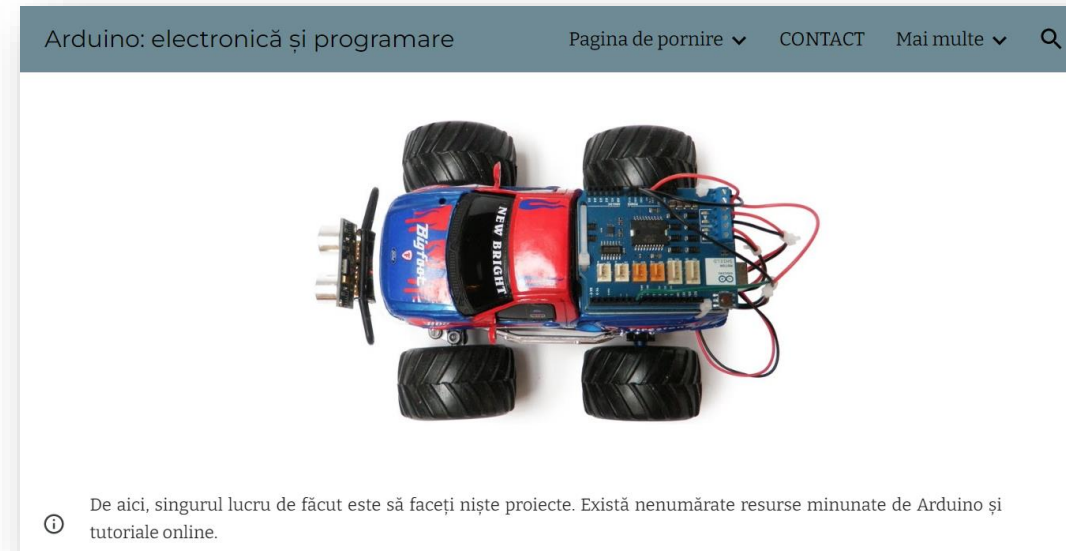
Arduino Tutorials - Examples
Bluino Electronics Educație ★★★★★ 1.720
E Toți
Conține anunțuri · Oferă achiziții în aplicație
i Aplicația este disponibilă pentru toate dispozitivele tale
A adaugă în lista de dorințe **Instalează**

1. Basic
Analog Read Serial
Bare Minimum
Blink
Digital Read Serial
Fade
Read Analog Voltage

2. Digital
3. Analog
4. Communication
5. Control Structures
6. Sensors
7. Display
8. Servos

Bare Minimum
This example contains the bare minimum of code you need for a sketch to compile properly on Arduino Software (IDE): the setup() method and the loop() method.
Hardware Required
• Arduino or Genuino Board
Circuit
Only your Arduino or Genuino Board is needed for this example.

```
void setup() {  
  Serial.begin(9600);  
}  
  
void loop() {  
  // AnalogReadSerial  
  // Read an analog input on pin 0, print the result  
  // (graphical representation is available using serial  
  // Attach the center wiper of a potentiometer to pin 0.  
  // This example code is in the public domain.  
  
  // The setup routine runs once when you press reset  
  void setup() {  
    // initialize serial communication at 9600 bits per  
    Serial.begin(9600);  
  }  
  
  // the loop routine runs over and over again forever  
  void loop() {  
    // read the input on analog pin 0:  
    int sensorValue = analogRead(A0);  
    // print out the value of the sensor:  
    Serial.print(sensorValue);  
    // delay 100ms between reads for the  
    delay(100);  
  }  
}
```



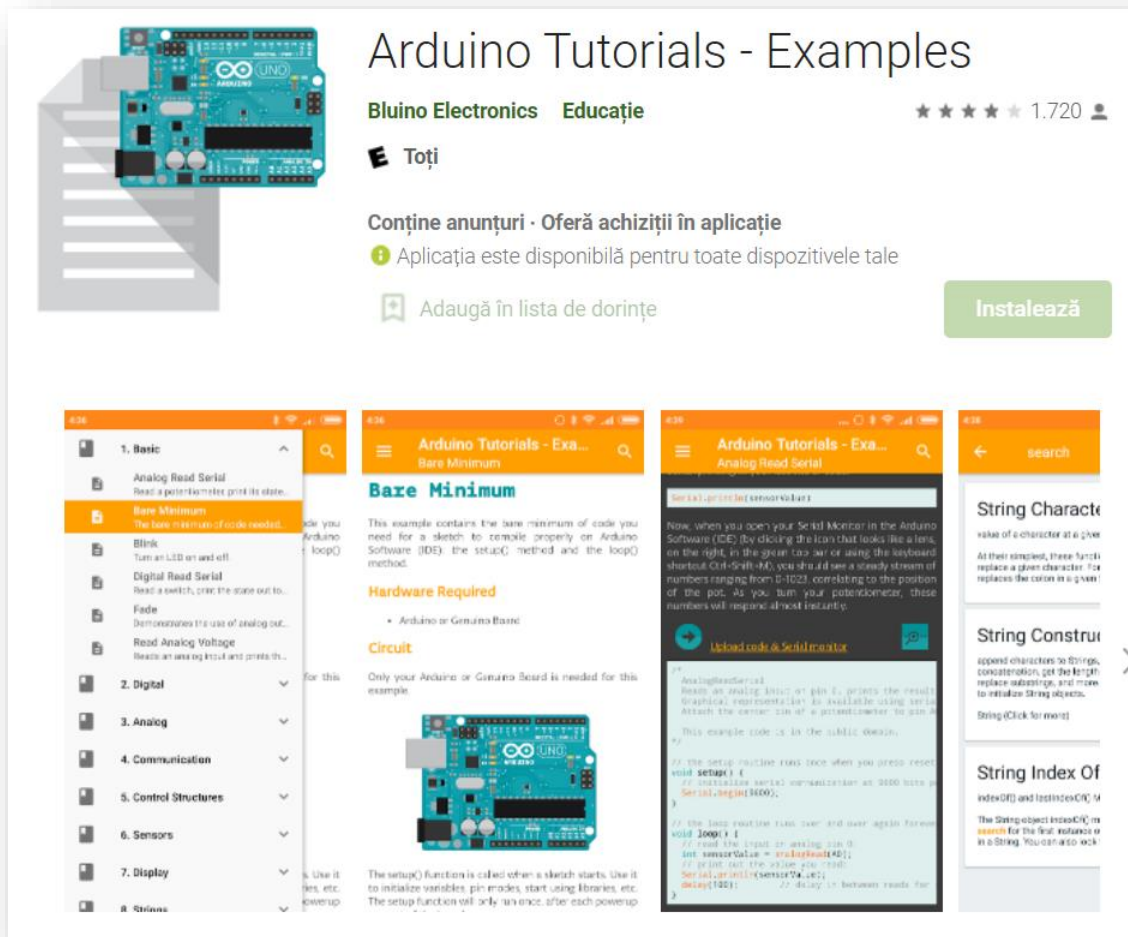
Arduino: electronică și programare Pagina de pornire CONTACT Mai multe

De aici, singurul lucru de făcut este să faceți niște proiecte. Există nenumărate resurse minunate de Arduino și tutoriale online.

<https://sites.google.com/site/arduinoelectronica/siprogramare/prima-lectie/5-introducere-in-arduino>

13.2. Aplicații cu Arduino

Tutoriale Arduino



Arduino Tutorials - Examples
Bluino Electronics Educație ★★★★★ 1.720

Toți

Conține anunțuri · Oferă achiziții în aplicație

Aplicația este disponibilă pentru toate dispozitivele tale

Adaugă în lista de dorințe **Instalează**

1. Basic

- Analog Read Serial
- Bare Minimum**
- Blink
- Digital Read Serial
- Fade
- Read Analog Voltage

2. Digital

3. Analog

4. Communication

5. Control Structures

6. Sensors

7. Display

8. Servos

Bare Minimum
This example contains the bare minimum of code you need for a sketch to compile properly on Arduino Software (IDE): the setup() method and the loop() method.

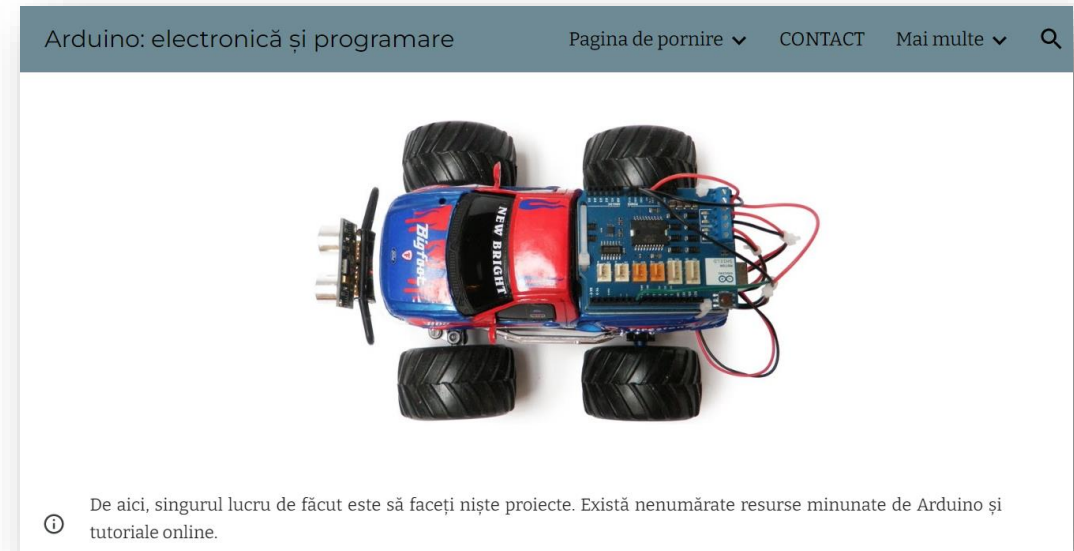
Hardware Required

- Arduino or Genuino Board

Circuit

Only your Arduino or Genuino Board is needed for this example.

```
void setup() {  
  Serial.begin(9600);  
}  
  
void loop() {  
  // AnalogReadSerial  
  // Read an analog input on pin 0, print the result  
  // (graphical representation is available using serial  
  // Attach the center wiper of a potentiometer to pin 0.  
  // This example code is in the public domain.  
  
  // The setup routine runs once when you press reset  
  void setup() {  
    // initialize serial communication at 9600 bits per  
    Serial.begin(9600);  
  }  
  
  // the loop routine runs over and over again forever  
  void loop() {  
    // read the input on analog pin 0:  
    int sensorValue = analogRead(A0);  
    // print out the value of the sensor:  
    Serial.print(sensorValue);  
    // delay in between reads for the  
    delay(1000);  
  }  
}
```



Arduino: electronică și programare

Pagina de pornire CONTACT Mai multe

De aici, singurul lucru de făcut este să faceți niște proiecte. Există nenumărate resurse minunate de Arduino și tutoriale online.

<https://sites.google.com/site/arduinoelectronica/siprogramare/prima-lectie/5-introducere-in-arduino>