

Tesztkérdések - Arduino programozás

A kérdések listája

1. Melyik nem C adattípus?

- char
- float
- long
- real

2. Melyik nem C adattípus?

- integer
- unsigned char
- int
- double

3. Melyik nem C adattípus?

- INT
- char
- double
- float

4. Egy előjeles 8 bites egész melyik mennyiség tárolására nem alkalmas?

- Kültéri hőmérséklet °C egységekben és felbontással.
- Relatív páratartalom % egységekben és felbontással.
- Autók gumibroncsának légnyomása 0.1 bar felbontással.
- Autó sebessége km/h egységekben és felbontással.

5. Egy előjel nélküli 16 bites egész melyik mennyiség tárolására nem alkalmas?

- 10 órányi időtartam másodpercekben
- A Föld átmérője km egységekben és felbontással.
- A szegedi lakosok száma.
- Repülők utazási magassága méter egységekben és felbontással.

6. Egy float típusú változó melyik mennyiség tárolására nem alkalmas 6 számjegynyi pontossággal?

- a Nap nyugalmi energiájának SI egységekben adott értéke.
- az Avogadro-állandó számértéke
- a Boltzmann állandó SI egységekben adott értéke
- az elemi töltés SI egységekben adott értéke

7. Melyik állítás téves?

- A double lebegőpontos típus Arduinón a float típussal azonos.
- A lebegőpontos számítások csak közelítő pontosságúak.
- A float lebegőpontos típus 4 bájt hosszúságú.
- Előjeles és előjel nélküli lebegőpontos típus is létezik.

8. Melyik állítás téves?

- Változó neve tartalmazhat _ karaktert.
- Változó neve tartalmazhat kis- és nagybetűket is.
- Változó neve nem tartalmazhat - karaktert.
- Változó neve kezdődhet számjeggyel.

9. Melyik változónév nem megfelelő?

- number_of_averages
- void
- _temporary
- referenceVoltage

10. Melyik nem C kulcsszó?

- while
- for
- function
- void

11. Melyik állítás téves?

- x és X különböző változóneveknek tekinthetők.
- A változók és függvények neve nem lehet while.
- Függvénynek és a benne deklarált változónak a neve lehet azonos.
- Változók neve nem kezdődhet számjeggyel.

12. Lehet-e két változónak azonos nevet adni?

- Semmilyen esetben sem lehet.
- Lehet, ha különböző blokkokban vannak deklarálni.
- Lehet, ha az egyik static, a másik nem.
- Lehet, ha különböző a változók típusa.

13. Melyik változódeklaráció nem alkalmas a és b létrehozására?

- `int a = 1; b=2;`
- `int a = 1;`
`int b;`
- `int a = 1, b=2;`
- `int a, b;`

14. Melyik változódeklaráció helytelen?

- `uint8_t a = 256;`
- `float a = 1E-6;`
- `int a = 10000;`
- `char a = -16;`

15. Melyik állítás téves az alábbi változók memóriaigényével kapcsolatban?

- `float x[10] = { 0.1, 1E-2 }; // 40 bytes`
- `int x[] = { -10, 10 }; // 4 bytes`
- `char x[] = { "0123456789" }; // 10 bytes`
- `uint16_t x[16]; // 32 bytes`

16. Melyik állítás téves a kommentben az alábbi változók memóriaigényével kapcsolatban?

- `char x[] = { "text" }; // 5 bytes`
- `float x[2]; // 8 bytes`
- `uint8_t x[16] = { 12, 24 }; // 2 bytes`
- `uint32_t x[16]; // 64 bytes`

17. Arduino UNO adatmemóriája 2048 byte méretű, aminek hozzávetőlegesen 60%-át ajánlott saját változók számára igénybe venni. Az alábbiak közül melyik sérti ezt a feltételt?

- `unsigned char x[1024];`
- `int x[512] = { -10, 10 };`
- `int32_t x[256];`
- `float x[400];`

18. Hőmérsékleti adatokat kezelünk egy programmal. A tartomány $-40^{\circ}\text{C}..85^{\circ}\text{C}$, a felbontás 1°C . Milyen változótípust célszerű választani?

- `uint8_t`
- `int`
- `long`
- `int8_t`

19. 0 V..5 V méréstartományú 10-bites előjel nélküli ADC adatának tárolásához milyen változótípust célszerű választani?

- `double`
- `uint16_t`
- `unsigned char`
- `int`

20. Egy 32 elemű tömb indexeléséhez milyen változótípust célszerű választani?

- `float`
- `int`
- `long`
- `unsigned char`

21. Egy program az alábbi deklarációt tartalmazza:

```
unsigned char c;
```

Melyik állítás igaz?

- c értéke 0 és 256 közötti egész szám lehet.
- c értéke 0 és 255 közötti egész szám lehet.
- c és C ugyanazt a változót jelenti.
- c nem szerepelhet aritmetikai műveletekben, például összeadásban.

22. Egy tömböt hozunk létre az alábbi módon:

```
char text[] = { "hello" };
```

Melyik állítás téves?

- A tömb 4 elemű.
- A tömb 5 elemű.
- `x[4]` értéke 0
- `x[0]` értéke 'h'

23. Egy tömböt hozunk létre az alábbi módon:

```
int x[4] = { 2, 4, 6, 8 };
```

Melyik állítás téves?

- az `x[1]++` utasítás legális.
- `x[2]` értéke 4
- `x[0]` értéke 2
- a tömbindex 0 és 3 közötti egész lehet.

24. A `setup()` függvényben egy `x`, a `loop()` függvényben egy `y` nevű változót deklarálunk. Melyik állítás igaz?

- `x` mindkét függvényből elérhető
- `x` nem érhető el a `loop()` függvényben és `y` nem érhető el a `setup()` függvényben.
- `x` mindkét függvényből elérhető, ha a `setup()` a `loop()` előtt van a kódban.
- `y` mindkét függvényből elérhető

25. Az `f()` függvényben egy `x`, a `g()` függvényben egy `y` nevű változót deklarálunk. Melyik állítás igaz?

- `x` akkor érhető el a `g()` függvényben, ha a forráskódban `f()` előbb szerepel, mint `g()`.
- `x` nem érhető el a `g()` függvényben és `y` nem érhető el a `f()` függvényben.
- `x` akkor érhető el a `g()` függvényben, ha `g()`-t `f()`-en belül hívjuk meg.
- `x` és `y` is mindkét függvényben elérhető.

26. Egy program az alábbi `loop()` függvényt tartalmazza:

```
loop()
{
  int i = 0;
  static int j = 0;

  i++;
  j++;
  Serial.print(i);
  Serial.print(" ");
  Serial.println(j);
}
```

A `loop` első három meghívásakor mit küld el ez a program a soros porton?

- 0 0
1 1
2 2
- 0 0
0 1
0 2
- 1 1
2 2
3 3
- 1 1
1 2
1 3

27. Egy program az alábbi `loop()` függvényt tartalmazza:

```
loop()
{
  int i;

  i++;
  Serial.println(i);
}
```

Mit küldd el ez a program a soros porton?

- 0-től kezdődően növekvő egész számokat.
- Határozatlan értéket, mert i-nek nincs megadva kezdőértéke.
- Mindig 1-et
- 1-től kezdődően növekvő egész számokat.

28. C utasításokat hogyan foglalhatunk egy blokkba?

- /* és */ jelek közé írva
- { } zárójelekkel
- begin és end szavak közé írva
- [] zárójelekkel

29. Milyen operátorral vizsgálhatjuk két változó egyenlőségét?

- ==
- ==
- =
- eq

30. Egy kód részlete az alábbi:

```
int x=8;
```

```
int y=3;
```

Melyik kifejezés logikai értéke hamis?

- (x / y) > (7 / 3)
- x % y == 2
- x & y == 0
- x && y

31. Egy kód részlete az alábbi:

```
float x=1.1;
```

```
float y=3.3;
```

```
if ( y / x == 3 ) switch_the_led_on();
```

Melyik állítás igaz?

- Nullával osztás történik, mert az x == 3 értéke 0.
- A feltételes utasítás nem biztosan hajtódik végre.
- A feltételes utasítás biztosan nem hajtódik végre.
- A feltételes utasítás biztosan végrehajtott.

32. Hogyan helyezhetők el megjegyzések egy C programban?

- szürke színű szöveggel
- /* és */ karakterpárok között
- () zárójelek között
- * és \ karakterpárok között

33. Melyik állítás téves?

- Csak egysoros megjegyzéseket használhatunk.
- Egysoros megjegyzés kezdődhet ugyanabban sorban, amiben kód is van.
- A megjegyzések nem csak angol nyelvűek lehetnek.
- A megjegyzések tartalmazhatnak akár C forráskódnak megfelelő szöveget is.

34. Melyik programsor nem felel meg helyes C kifejezésnek?

- `x = a \ b;`
- `x = a ^ b;`
- `x = a % b;`
- `x = a >> b;`

35. Melyik utasítás számítja ki az x változó négyzetét?

- `x*x;`
- `x^2;`
- `x >> 2;`
- `x ** 2;`

36. Melyik kód számítja ki helyesen a másodfokú egyenlet gyökeit?

- `x1=(-b+sqrt(b*b-4*a*c))/2/a;`
`x2=(-b-sqrt(b*b-4*a*c))/2/a;`
- `x1=-b+sqrt(b*b-4*a*c)/2*a;`
`x2=-b-sqrt(b*b-4*a*c)/2*a;`
- `x1=-b+pow(b*b-4*a*c,0.5)/2*a;`
`x2=-b-pow(b*b-4*a*c,0.5)/2*a;`
- `x1=(-b+sqrt(b*b-4*a*c))/2*a;`
`x2=(-b-sqrt(b*b-4*a*c))/2*a;`

37. Melyik kód nem számítja ki helyesen a kívánt értéket?

- párhuzamosan kötött ellenállások eredője: $1/(1/r1+1/r2)$;
- derékszögű háromszög átfogója: `sqrt(a*a+b*b)` ;
- téglalap területe osztva a kerületével: $a*b/(a+b)/2$;
- kör sugara a terület ismeretében: $k/2*pi$;

38. Melyik programsor számítja ki egy változó négyzetgyökét?

- `x^0.5;`
- `root(x);`
- `sqrt(x);`
- `x**(1/2);`

39. Melyik állítás téves?

- `sqrt(x)` az x változó négyzetgyökét számítja ki.
- `floor(x)` az x változó egészrészét számítja ki.
- `log(x)` az x változó tízes alapú logaritmusát számítja ki.
- `pow(x,3)` az x változó köbét számítja ki.

40. Melyik állítás téves?

- Az $x = a*b+c$; és az $x = c+a*b$; ekvivalens.
- Ha x unsigned char típusú változó és értéke 1, akkor -x értéke 254.
- Ha x és y egész típusú változó, akkor a hányadosuk is egész a legközelebbi egészre kerekítve.
- Két változó egyenlőségét az == operátorral ellenőrizhetjük.

41. Melyik C adattípus tartománya nincs helyesen van megadva?

- unsigned char: $0..2^8-1$
- int16_t: $-2^{15}..2^{15}$
- char: $-128..127$
- uint32_t: $0..2^{32}-1$

42. Az Arduino A/D konverter referenciafeszültségét az analogReference(DEFAULT) függvényhívással állítjuk be. Melyik állítás téves ideális esetet tekintve?

- 2.5 V bemenő feszültség esetén az analogRead függvény visszatérési értéke 511
- 0 V bemenő feszültség esetén az analogRead függvény visszatérési értéke 0
- A felbontás értéke $5\text{ V}/2^{10}$.
- 1.25 V bemenő feszültség esetén az analogRead függvény visszatérési értéke 256

43. Arduino áramkörben az A/D konverter 10-bites, a referenciafeszültség 5 V. A bemeneti feszültségtartomány $0\text{ V}..V_{\text{ref}}$. A kimeneti adat előjel nélküli egész szám a 0 és 1023 közötti tartományban. A mért feszültséget melyik C forráskód adja vissza helyesen?

- `voltage = adc_data * 5.0 / 1023;`
- `voltage = adc_data * 5.0 / 512;`
- `voltage = adc_data * 5.0 / 255;`
- `voltage = adc_data * 5.0 / 1024;`

44. Egy program az alábbi kódrészletet tartalmazza:

```
float x = 7/4;
```

Mennyi lesz x értéke?

- a fordító hibát jelez erre a kódra
- 1.75
- 2.0
- 1.0

45. Egy program az alábbi kódrészletet tartalmazza:

```
float x = 3/4;
```

Mennyi lesz x értéke?

- 1.0
- a fordító hibát jelez erre a kódra
- 0.75
- 0.0

46. Egy program az alábbi kódrészletet tartalmazza:

```
int x;
```

```
x = 7.0/4.0;
```

Mennyi lesz x értéke?

- 0
- 7
- 2
- 1

47. Egy program az alábbi kódrészletet tartalmazza:

```
if ( (a > 1) && (a < 10) ) x = 1;
```

```
else x = -1;
```

Melyik állítás téves?

- Ha a értéke 5, akkor x értéke 1 lesz.

- Ha a értéke 1, akkor x értéke 1 lesz.
- Ha a értéke 11, akkor x értéke -1 lesz.
- Ha a értéke 0, akkor x értéke -1 lesz.

48. Egy logikai kifejezés az alábbi:

`(x < 1) && (y > 2)`

Melyik ennek a negáltja?

- `(x < 1) || (y > 2)`
- `(x >= 1) || (y <= 2)`
- `(x >= 1) && (y <= 2)`
- `(x > 1) && (y < 2)`

49. Egy logikai kifejezés az alábbi:

`(x == 1) || (y != 2)`

Ha ez egy if utasításban van, mikor hajtódik végre az else ág?

- ha y értéke 2, x értéke ekkor már nem számít
- ha x értéke nem 1, y értéke ekkor már nem számít
- ha x értéke nem 1 és y értéke 2
- ha x értéke nem 1 és y értéke nem 2

50. Egy program az alábbi kódrészletet tartalmazza:

`if (a && b || c)`

Mikor nem hajtódik végre a feltételes utasítás?

- Ha a = 0, b = 1, c = 1
- Ha a = 1, b = 1, c = 0
- Ha a = 1, b = 0, c = 1
- Ha a = 0, b = 1, c = 0

51. Melyik programsor helyes C forráskód?

- `for(i=1; i<10; i*=2)`
- `while (i=0; i<10; i++)`
- `while if (i<10)`
- `for(i=0, i<10, i++)`

52. Melyik ciklus magja fut le legalább egyszer?

- `i=5;`
`do`
`{`
`i--;`
`} while (i>10);`
- `i=5;`
`while (i<5) i++;`
- `for(i=10, i<10, i++);`
- `for(i=5, i<0, i--);`

53. Az alábbi ciklusok változói a ciklusmagban bizonyos értékeket vesznek fel. Melyik állítás téves? A változók deklarációja:

`int i;`

`float x;`

- `for(i=1; i<10; i*=2) : kettő hatványai 1-től 8-ig`

- `for(i=10; i>0; i--)` : egész számok 10-től 0-ig csökkenő sorrendben
- `for(x=1.0; x<100; x*=1.1)` : mértani sor 1-től, 1.1 kvócienssel
- `for(x=-1.0; x<1.05; x+=0.1)` : számtani sor -1-től, 1-ig, 0.1 lépésekben

54. Egy program az alábbi kódrészletet tartalmazza:

```
x = 10;
y = 10;
while ( x < 10 ) y = y + x;
```

Mennyi lesz y értéke a kódrészletet lefutása után?

- 10
- 100
- 20
- 110

55. Egy program az alábbi kódrészletet tartalmazza:

```
x = 0;
for(i=0; i<10; i++)
{
    x = x + 1;
}
```

Mennyi lesz x értéke a kódrészletet lefutása után?

- 10
- 11
- 50
- 9

56. Egy program az alábbi kódrészletet tartalmazza:

```
x = 0;
for(i=0; i<10; i++)
{
    x = x + i;
}
```

Mit számít ki a program?

- A 0 és 10 közötti egész számok összegét.
- Az 1 és 10 közötti egész számok összegét.
- Az 1 és 9 közötti egész számok összegét.
- A 0 és 9 közötti egész számok összegét.

57. Egy program az alábbi kódrészletet tartalmazza:

```
for(i=0; i<3; i++)
{
    Serial.println(i);
}
Serial.println(i);
```

Milyen számokat ír ki ez a kódrészlet?

- 0
1
2
0
- 0
1
2

- 0
- 1
- 2
- 2
- 0
- 1
- 2
- 3

58. Egy program az alábbi kódrészletet tartalmazza:

```
x = 0;
for(i=0; i<10; i=i+2)
{
    x = x + 1;
}
```

Hányszor fut le a ciklusmag?

- 10
- 5
- 6
- 4

59. Egy program az alábbi kódrészletet tartalmazza:

```
x = 0;
for(i=0; i<=10; i=i+5)
{
    x = x + 1;
}
```

Hányszor fut le a ciklusmag?

- 2
- 10
- 3
- 11

60. Egy program az alábbi kódrészletet tartalmazza:

```
x = 0;
for(i=0; i<10; i=i*2)
{
    x = x + 1;
}
```

Hányszor fut le a ciklusmag?

- Ez egy végtelen ciklus.
- 4
- Ez egy hibás kód, a for ciklus harmadik eleme nem lehet $i=i*2$
- 5

61. Egy program az alábbi kódrészletet tartalmazza:

```
for(i=0; i<4; i++)
{
    s=0;
    for(j=0; j<i; j++) s+=j;
    Serial.println(s);
}
```

Milyen számokat ír ki ez a kódrészlet?

- 0
1
3
6
- 1
2
3
4
- 1
3
6
10
- 0
1
2
3

62. Melyik állítás téves egy Arduino program esetén?

- A loop() függvény egy végtelen ciklusban van, folyamatosan ismétlődik.
- A kód nem tartalmaz main() függvényt.
- Ha a setup() és loop() függvény lefutott, a program futása befejeződik.
- A setup() függvény csak a program elején fut le egyszer.