



TANÁRTOVÁBBKÉPZÉS 2026



**Az Európai Unió
támogatásával**

Az Európai Unió finanszírozásával. Az itt szereplő információk és állítások a szerző(k) álláspontját képviselik, és nem feltétlenül tükrözik az Európai Unió vagy a Tempus Közalapítvány hivatalos véleményét. Sem az Európai Unió, sem a támogatást nyújtó hatóság nem vonható felelősségre miattuk. Ez a projekt az Erasmus+ KA210-SCH program támogatásával valósul meg.



PÁZMÁNY | ITK

Digitális kultúra

TANÁRTOVÁBBKÉPZÉS

2026

PROGRAMOZÁS OKTATÁSA MOTIVÁLT ÉS NEM MOTIVÁLT TANULÓKKAL

2026. április 22

Siegler Gábor

tanár, Berzsenyi Dániel Gimnázium

SEGÉDANYAGOK

- feladatok, források
- segédanyagok, útmutatók
- félkész megoldások

<https://tinyurl.com/bdh62run>

KERETTANTERV 9-10. ÉVFOLYAMON

A témakörök áttekintő táblázata:

| Témakör neve | Javasolt óraszám |
|--|-------------------------|
| Algoritmizálás, formális programozási nyelv használata | 25 |
| Információs társadalom, e-Világ | 3 |
| Mobiltechnológiai ismeretek | 4 |
| Szövegszerkesztés | 11 |
| Számítógépes grafika | 14 |
| Multimédiás dokumentumok készítése | 4 |
| Online kommunikáció | 4 |
| Publikálás a világhálón | 14 |
| Táblázatkezelés | 12 |
| Adatbázis-kezelés | 5 |
| A digitális eszközök használata | 6 |
| Összes óraszám: | 102 |



TÉMAKÖR: Algoritmizálás, formális programozási nyelv használata

JAVASOLT ÓRASZÁM: 25 óra

TANULÁSI EREDMÉNYEK**A témakör tanulása hozzájárul ahhoz, hogy a tanuló a nevelési-oktatási szakasz végére:**

- érti az egyszerű problémák megoldásához szükséges tevékenységek lépéseit és kapcsolatukat;
- ismeri a következő elemi adattípusok közötti különbségeket: egész, valós szám, karakter, szöveg, logikai;
- ismeri az elemi és összetett adattípusok közötti különbségeket;
- érti egy algoritmusleíró eszköz alapvető építőelemeit;
- érti a típusalgoritmusok felhasználásának lehetőségeit.

A témakör tanulása eredményeként a tanuló:

- példákban, feladatok megoldásában használja egy formális programozási nyelv fejlesztői környezetének alapszolgáltatásait;
- szekvencia, elágazás és ciklus segítségével algoritmust hoz létre, és azt egy magas szintű formális programozási nyelven kódolja;
- a feladat megoldásának helyességét teszteli;
- tapasztalatokkal rendelkezik hétköznapi jelenségek számítógépes szimulációjáról;
- hétköznapi, oktatáshoz készült szimulációs programokat használ;
- tapasztalatokat szerez a kezdőértékek változtatásának hatásairól a szimulációs programokban.

**FEJLESZTÉSI FELADATOK ÉS ISMERETEK**

- Az algoritmikus gondolkodást segítő informatikai eszközök és szoftverek használata
- Hétköznapi tevékenységekből a folyamat és az adatok absztrakciója
- A problémamegoldó tevékenység tervezési és szervezési kérdései
- A problémamegoldáshoz tartozó algoritmusok megismerése. Algoritmus leírása egy lehetséges módjának megismerése
- Az algoritmus végrehajtásához szükséges adatok és az eredmények kapcsolatának vizsgálata
- Az elemi adatok és sorozatok megkülönböztetése, kezelése és használata
- Szekvencia, elágazások és ciklusok
- Példák típusalgoritmus használatára
- A vezérlési szerkezetek megfelelői egy formális programozási környezetben
- Elágazások, feltételek kezelése, többirányú elágazás, ciklusok
- Változók, értékadás. Eljárások, függvények alkalmazása
- A program megtervezése, kódolása, tesztelése
- Az objektumorientált szemlélet megalapozása
- Mások által készített alkalmazások paramétereinek a program működésére gyakorolt hatásának vizsgálata

KERETTANTERV 9-10. ÉVFOLYAMON

FOGALMAK

algoritmuselemek, tervezési folyamat, adatok absztrakciója, algoritmusleírási mód, szekvencia, elágazás, ciklus, egész szám, valós szám, karakter, szöveg, sorozat, logikai adat, egyszerű algoritmusok tervezése, vezérlési szerkezetek, eljárás, függvény, kódolás, típusfeladatok, tesztelés, hibajavítás



JAVASOLT TEVÉKENYSÉGEK

- Egy formális programozási nyelv megismerése közösen megoldott egyszerű példákon keresztül
- Típusok, változók és vezérlőszervezetek (szekvencia, elágazás, ciklus) tudatos választását igénylő feladatok önálló megoldása, a választás indoklása
- Programozási feladatok megoldása során algoritmusok megismerése, leírása és kódolása
- Az algoritmusok és az adatszerkezetek kapcsolatának használatát igénylő programozási feladatok megoldása, a választás indoklása
- Konkrét programozási feladathoz kapcsolódó algoritmusok leírása egy lehetséges módszerrel
- Feladat megoldása során a fejlesztői környezet lehetőségeinek használata (pl. tesztelés)
- **Feladatmegoldás strukturálatlan algoritmussal** és függvények, eljárások használatával
- Olyan problémák közös megoldása, amelyek során a függvények, eljárások paraméterezése a paraméterátadás különböző típusainak alkalmazását igényli
- Egy saját vagy más által készített program tesztelése
- Adott feladathoz készült különböző megoldások közös megbeszélése

MÓDSZERTANI KÉRDÉSEK

Sorrend korosztály szerint:

- Szerepjáték
- Robot
- Blokkprogramozás
- Automata-elvű programozási nyelv
- Formális programozási nyelv



- Melyik formális programozási nyelv?

TANKÖNYVEK



Kiegészítés: Programozás fejezetek c++ és c# nyelven <https://sztzs.infokatedra.hu/public/prog/>

NKP OKOSTANKÖNYVEK

<https://www.nkp.hu/okostankonyvek>

Válassz tankönyvet!

| Általános iskola | Középiskolák | | | | | | | | | | SNI | |
|------------------|--------------|----|----|----|----|----|----|----|----|-----|-----|-----|
| Évfolyam: | 1. | 2. | 3. | 4. | 5. | 6. | 7. | 8. | 9. | 10. | 11. | 12. |

| | | |
|--|--|---|
|  <p>Digitális kultúra 11-12. okosgyűjtemény (emelt szint)</p> |  <p>Digitális kultúra 11. (NAT2020)</p> |  <p>Digitális kultúra 9-12. okosgyűjtemény</p> |
|--|--|---|

NULLÁTÓL A TÉTELEKIG

- Saját gyűjtemény
- A gyűjtemény **folyamatos** karbantartása (jó, problémás, nehéz,... besorolások), bővítése
- Feladat füzérek, csomók, elágazások
- Tematika:

Kifejezések bevezetése

Ciklusok bevezetése

Karaktergrafika

Szövegkezelés

Vegyes feladatok ciklusra, elágazásra és szövegkezelésre

Vektor és tömb, összetett adatszerkezet bevezetése

Átvezetés a programozási tételekhez

PÉLDA

Megadott óra, perc, másodperc hármassal megadott időt váltsunk át másodpercre!

Sikeres megoldás után készítsük el a feladat megfordítottját!

Milyen szöveget zár be egymással kis és nagymutató adott időpontban?

Különböző sportágak játékideje – hogyan mérik a játékidőket?

Új fogalmak: beszédes változónevek, foglalt nevek, típusfüggő operátorok, /, %, div, mod

MOTIVÁCIÓS ESZKÖZÖK

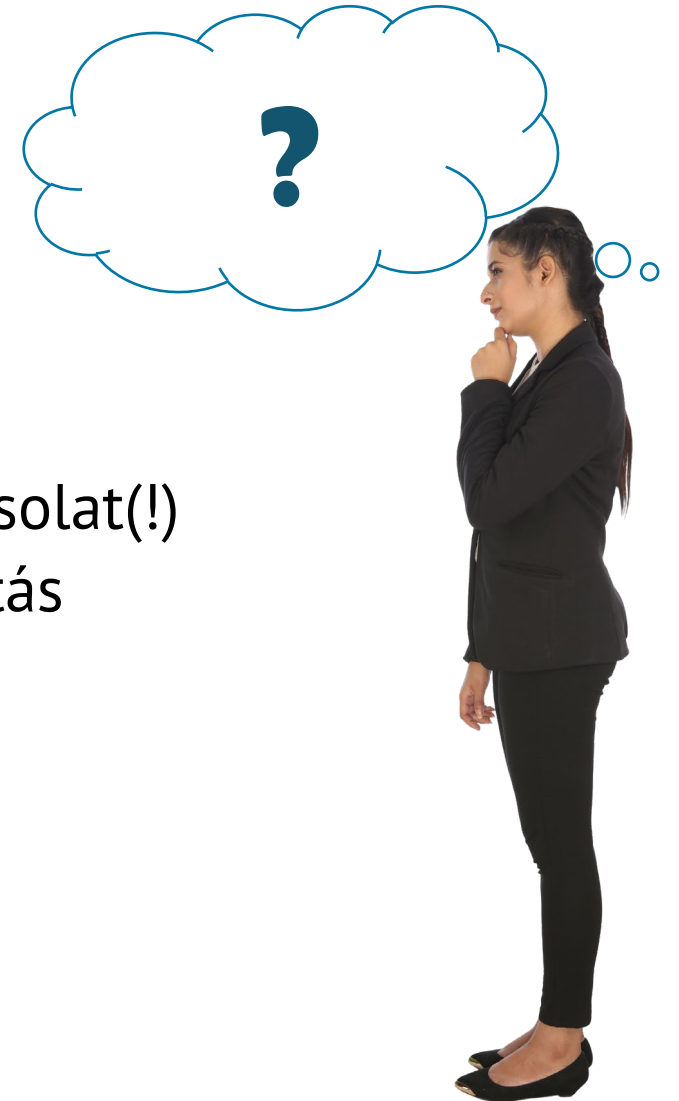
- Motivált és nem motivált tanulók más-más módszert kívánnak
- Motivált tanuló:
 - Ugyanannyi időre több és/vagy nehezebb feladat (feladatfüzér)
 - Több önállóság, kevesebb segítség – online dokumentációk használata
 - Előre tisztázni kell a követelményeket
 - Ne adjunk automatikusan jelest
 - Ne kapjon felmentést a diák az órák látogatása alól
 - Beszámolási kötelezettség
 - Verseny (?)
- Tanár szerepe

Problémák - személyiség típusok: csak annyit, amennyit muszáj, mindig ő az okosabb, nyerni akar, különben nem érdekes,...

MOTIVÁCIÓS ESZKÖZÖK

Nem motivált tanuló:

- Oka?
- Előző korosztályos lépések hiányoznak! – pótlás?
- A feladatok szituációja! – szövegértési problémák
- Teremrend, a számítógépek elhelyezése – egyéni kapcsolat(!)
- Tábla(?), prezentáció(?), párhuzamos munka(?) – aktivitás
- Ne 1 évben, hanem több évre elosztva haladjunk
- Változtatni: módszereket, forrásokat
- Tanári cselekvés – színészi attitűd szükséges



FELADATFŰZÉR PÉLDA

Írjunk programot, ami egy N magas számhegyet hoz létre!

Példa: Bemenet: 5 Kimenet: 1 2 3 4 5 4 3 2 1

Előző programunkat bővítsük azzal, hogy N magas számhegységet hozzunk létre!

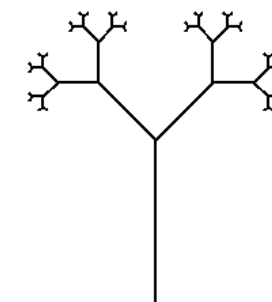
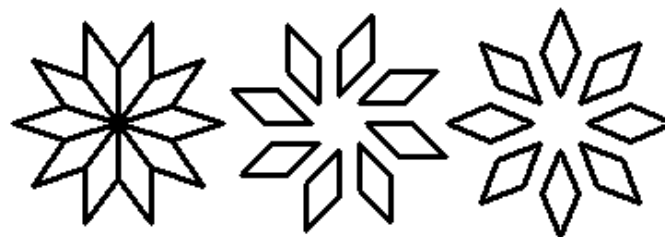
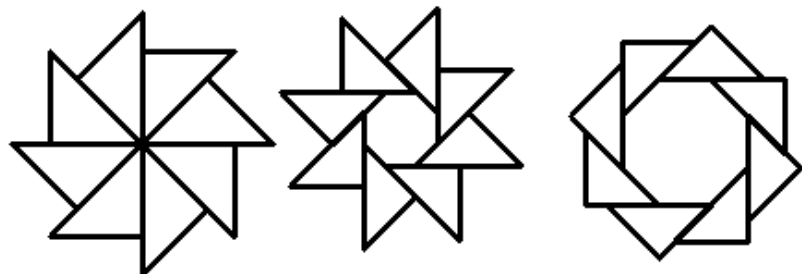
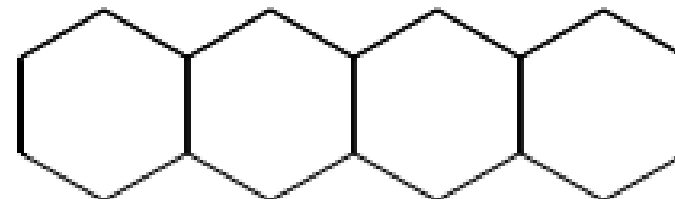
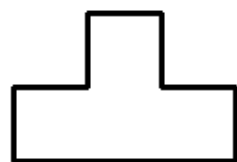
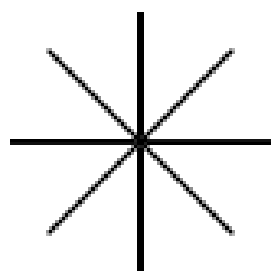
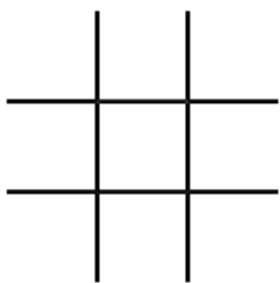
Példa: Bemenet: 5 Kimenet: 1 2 3 4 5 4 3 2 1 1 2 3 4 3 2
1 1 2 3 2 1 1 2 1 1

Írjunk programot/függvényt, ami egy N magas számlépcsőt hoz létre!

Példa: Bemenet: 5 Kimenet: 1 2 2 3 3 3 4 4 4 4 5 5 5 5 5

TÖBB ESZKÖZZEL MEGOLDHATÓ FELADATOK

Blokkprogramozás és Vektorgrafika



TÖBB ESZKÖZZEL MEGOLDHATÓ FELADATOK

Táblázatkezelés és programozás

- hatvány
- számhegy, számhegység
- Newton-iteráció
- faktoriális
- Fibonacci-számok
- karaktergrafika: négyzet, háromszög, átló
- szorzótábla
- gondoltam egy számot, ...

TÖBB ESZKÖZZEL MEGOLDHATÓ FELADATOK

Motivált tanulóknál:

- Szimulációk (Átszivárgás, Lámpák, Sejtautomaták,...)
- Adatbázis és programozás

Emelt érettségi feladatok:

- programozás kiegészítve táblázatkezeléssel
- programozás kiegészítve html táblázat kimenettel

PROGRAMOZÁS KÖVETELMÉNYEK KÖZÉP- ÉS EMELT SZINTŰ GYAKORLATI ÉRETTSÉGIN

2026. április 22

Siegler Gábor

tanár, Berzsenyi Dániel Gimnázium

HASONLÓSÁGOK ÉS KÜLÖNBSÉGEK

7. Algoritmizálás, adatmodellezés

| TÉMÁK | VIZSGASZINTEK | |
|--|--|---|
| | Középszint | Emelt szint |
| 7.1. Adatszerkezetek, adat be- és kivitel, 7.1.1. Egyszerű adattípusok 7.1.2. Összetett adattípusok | Tudja használni az egyszerű adattípusokat: egész és valós szám, logikai érték, karakter. Ismerje az egyszerű adattípusokon végezhető műveleteket. Tudjon véletlen értékeket előállítani. Tudja használni a szöveg és a sorozat elemi típusokból felépülő összetett adattípusokat. Ismerje az ezeken az adattípusokon végezhető műveleteket. | Legyen képes többdimenziós adatszerkezet használatára. Tudjon összetett adatszerkezetben különböző típusú adatokat tárolni. |
| 7.2. Algoritmisleíró eszközök 7.2.1. Feladatmegoldás egy algoritmisleíró eszköz segítségével | Ismerje az algoritmusok mondatszerű leírását. | Legyen képest megadott algoritmust módosítani. |
| 7.3. Elemi algoritmusok 7.3.1. A programozás alapelvei | Feladatmegoldásához tudja használni az alapvető programszerkezeteket: szekvencia, elágazás, ciklus. | Tudja használni a kiválogatás, rendezés, szétválogatás, metszet és unió típusalgoritmusait. |

HASONLÓSÁGOK ÉS KÜLÖNBSÉGEK

| TÉMÁK | VIZSGASZINTEK | |
|--|---|---|
| | Középszint | Emelt szint |
| | Tudja használni a típusalgoritmusokat: összegzés, eldöntés, kiválasztás, keresés, megszámlálás, maximum-kiválasztás. Tudjon egyetlen típusalgoritmust igénylő problémát megoldani. | Tudjon közepes nehézségű, típusalgoritmusok egymásba építését igénylő, összetett problémát megoldani. |
| 7.4. Rekurzió 7.4.1. Rekurzív algoritmus | | Ismerje a rekurzió fogalmát. |

8. A programozás eszközei

| TÉMÁK | VIZSGASZINTEK | |
|--|---|--|
| | Középszint | Emelt szint |
| <p>8.1. Programozási nyelv</p> <p>8.1.1. Egy programozási nyelv ismerete</p> <p>8.1.2. Adat be- és kivitel</p> | <p>Ismerje egy programozási nyelven a változódeklaráció, a be- és kiviteli utasítások; alapvető programszerkezetek, azaz szekvencia, elágazás, ciklus megvalósítását.</p> <p>Legyen képes egy mondatszerű leírással készült algoritmust a használt programozási nyelvben kódolni.</p> <p>Legyen képes az adott problémát megoldó programot írni.</p> <p>Tudjon olyan parancssori alkalmazást készíteni, amely billentyűzetről képes adatokat olvasni, illetve monitoron megjeleníteni.</p> <p>Tudjon a felhasználóval kulturáltan kommunikáló adatbevitelt és adatkivittelt írni.</p> | <p>Ismerje egy programozási nyelven az eljárások, függvények; állománykezelő műveletek megvalósítását.</p> <p>Legyen képes mondatszerű leírással megadott rekurzív algoritmust kódolni, és felhasználni.</p> <p>Legyen képes a szöveges állományokra alkalmazható műveleteket megvalósítani.</p> |
| <p>8.2.1. Kódolási, szerkesztési eszközök valamilyen programnyelvi fejlesztői környezetben</p> <p>8.2.2. Tesztelés</p> | <p>Ismerje egy, a választott programozási nyelvvel használható fejlesztőkörnyezet működését, használatát, eszközeit.</p> <p>Legyen képes a program különböző kimeneteinek tesztelésére alkalmas mintaadatokkal ellenőrizni a program helyes működését.</p> | <p>Tudjon nyomkövetéssel programot tesztelni.</p> |

FELADATOKRÓL

Tervezett megoldási idő és pontszám:

Középszint: 30 perc/180 perc

15 pont / 100 pont (**15,0%**)

Emelt szint: 100 perc/240 perc

50 pont / 120 pont (**41,6%**)

Név: osztály:.....

ÉRETTSÉGI VIZSGA • 2025. május 12.

DIGITÁLIS KULTÚRA

**KÖZÉPSZINTŰ
GYAKORLATI VIZSGA**

2025. május 12. 8:00

Időtartam: 180 perc

| | |
|---------------------------|--|
| Beadott dokumentumok | |
| Piszkozati pótlapok száma | |

Értékelésre az alábbi állományokat adom be:

| | |
|-------------------------------|--|
| Szövegszerkesztés | |
| Vizuális elemek | |
| Táblázatkezelés | |
| Adatbázis-kezelés | |
| Algoritmizálás és programozás | A program forráskódját tartalmazó állomány nevet adja meg! |

OKTATÁSI HIVATAL

Digitális kultúra
K2314

Nat2020

RÉSZLETES VIZSGAKÖVETELMÉNYEK 1.

7. Algoritmizálás, adatmodellezés

| TÉMÁK | VIZSGASZINTEK | |
|--|---|---|
| | Középszint | Emelt szint |
| 7.1. Adatszerkezetek, adat be- és kivitel, 7.1.1. Egyszerű adattípusok 7.1.2. Összetett adattípusok | <p>Tudja használni az egyszerű adattípusokat: egész és valós szám, logikai érték, karakter. Ismerje az egyszerű adattípusokon végezhető műveleteket. Tudjon véletlen értékeket előállítani.</p> <p>Tudja használni a szöveg és a sorozat elemi típusokból felépülő összetett adattípusokat. Ismerje az ezeken az adattípusokon végezhető műveleteket.</p> | <p>Legyen képes többdimenziós adatszerkezet használatára. Tudjon összetett adatszerkezetben különböző típusú adatokat tárolni.</p> |
| 7.2. Algoritmusleíró eszközök 7.2.1. Feladatmegoldás egy algoritmusleíró eszköz segítségével | <p>Ismerje az algoritmusok mondatszerű leírását.</p> | <p>Legyen képest megadott algoritmust módosítani.</p> |
| 7.3. Elemi algoritmusok 7.3.1. A programozás alapelvei | <p>Feladatmegoldásához tudja használni az alapvető programszerkezeteket: szekvencia, elágazás, ciklus.</p> <p>Tudja használni a típusalgoritmusokat: összegzés, eldöntés, kiválasztás, keresés, megszámlálás, maximum-kiválasztás.</p> <p>Tudjon egyetlen típusalgoritmust igénylő problémát megoldani.</p> | <p>Tudja használni a kiválogatás, rendezés, szétválogatás, metszet és unió típusalgoritmusait.</p> <p>Tudjon közepes nehézségű, típusalgoritmusok egymásba építését igénylő, összetett problémát megoldani.</p> |
| 7.4. Rekurzió 7.4.1. Rekurzív algoritmus | | <p>Ismerje a rekurzió fogalmát.</p> |



RÉSZLETES VIZSGAKÖVETELMÉNYEK 2.

8. A programozás eszközei

| TÉMÁK | VIZSGASZINTEK | |
|---|---|--|
| | Középszint | Emelt szint |
| 8.1. Programozási nyelv 8.1.1. Egy programozási nyelv ismerete 8.1.2. Adat be- és kivitel | <p>Ismerje egy programozási nyelven a változódeklaráció, a be- és kiviteli utasítások; alapvető programszerkezetek, azaz szekvencia, elágazás, ciklus megvalósítását.</p> <p>Legyen képes egy mondatszerű leírással készült algoritmust a használt programozási nyelvben kódolni.</p> <p>Legyen képes az adott problémát megoldó programot írni.</p> <p>Tudjon olyan parancssori alkalmazást készíteni, amely billentyűzetről képes adatokat olvasni, illetve monitoron megjeleníteni.</p> <p>Tudjon a felhasználóval kulturáltan kommunikáló adatbevitelt és adatkivitel írti.</p> | <p>Ismerje egy programozási nyelven az eljárások, függvények; állománykezelő műveletek megvalósítását.</p> <p>Legyen képes mondatszerű leírással megadott rekurzív algoritmust kódolni, és felhasználni.</p> <p>Legyen képes a szöveges állományokra alkalmazható műveleteket megvalósítani.</p> |
| 8.2. Programfejlesztés és környezet 8.2.1. Kódolási, szerkesztési eszközök valamilyen programnyelvi fejlesztői környezetben 8.2.2. Tesztelés | <p>Ismerje egy, a választott programozási nyelvvel használható fejlesztőkörnyezet működését, használatát, eszközeit.</p> <p>Legyen képes a program különböző kimeneteinek tesztelésére alkalmas mintaadatokkal ellenőrizni a program helyes működését.</p> | <p>Tudjon nyomkövetéssel programot tesztelni.</p> |

MINTÁK A KÖZÉPSZINTŰ FELADATSOR

A képernyőre írást igénylő részfeladatok esetén az ékezetmentes kiírás is elfogadott.

1. Olvassa be a mintának megfelelően, és tárolja el, hogy hány héten át tartott a fogyókúra, és Mari néni milyen célt tűzött ki maga elé!
2. Olvassa be a mintának megfelelően Mari néni tömegét a fogyókúra heteiben!
3. Elérte-e Mari néni a kitűzött célt? Ha igen, írassa ki a képernyőre az első olyan hét sorszámát, amikor Mari néni tömege már nem haladta meg a kitűzött célt! Ha egyetlen héten sem érte ezt el, akkor írassa ki: „Sajnos Mari néni nem érte el a célját.”
4. Mari néni nem tartja be következetesen az előírásokat, ezért előfordul, hogy a tömege egyik hétről a másikra nem csökken, hanem növekszik. A fogyókúra időszaka alatt hány olyan hét volt, amikor Mari néni tömege nőtt az előző héthez képest? Válaszát a mintának megfelelően írassa ki a képernyőre!

```
Hetek száma=6
Elérni kívánt testtömeg (kg)=93.5
1. héten=95.5
2. héten=94.3
3. héten=94.4
4. héten=93.3
5. héten=93.8
6. héten=92.9
Mari néni a(z) 4. héten érte el a célt.
A tömege 2 esetben nőtt egyik hétről a másikra.
```

15 pont



```
1 #include <iostream>
2 using namespace std;
3
4 int n;
5 double cel, db, tomeg[365];
6
7 int main() {
8     cout<<"Hetek szama="; cin>>n;
9     cout<<"Elerni kivant testtomeg (kg)="; cin>>cel;
10    for(int i=0;i<n;i++){
11        cout<<i+1<<".heten=";cin>>tomeg[i];
12    }
13    int j=0;
14    while (tomeg[j]>cel && j<n)
15        j++;
16    if(j<n)cout<<"Mari neni a "<<j+1<<". heten erte el a celt.";
17        else cout<<"Nem erte el a celt";
18    cout<<endl;
19    db=0;
20    for(int i=0;i<n-1;i++)
21        if(tomeg[i+1]>tomeg[i]) db++;
22    cout<<"A tomeg "<<db<<" esetben nott egyik hetrol a masikra."<<endl;
23    return 0;
24 }
25
```




```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     setlocale(LC_ALL, "hun");
6     int uveg[15]={5, 2, 2, 4, 3, 2, 4, 10, 5, 5, 3, 5, 4, 3, 3};
7     int osszeg=0, i, legnagyobb=0, L;
8     cout<<"2. feladat"<<endl;
9     cout<<"Mari néni lekvárja (dl): "; cin>>L;
10    for(int i=1; i<15; i++)
11        if(uveg[i]>uveg[legnagyobb]) legnagyobb=i;
12    cout<<"3. feladat"<<endl;
13    cout<<"A legnagyobb üveg: "<<uveg[legnagyobb]<<" dl és "<<legnagyobb+1<<" a sorban."<<endl;
14    for(int i=0; i<15; i++)
15        osszeg+=uveg[i];
16    cout<<"4. feladat"<<endl;
17    if(osszeg>=L) cout<<"Elegendő üveg volt."<<endl;
18    else cout<<"Maradt lekvár."<<endl;
19
20    return 0;
21 }
```

MINTÁK AZ EMELT SZINTŰ FELADATSOROKBÓL

| | A | M |
|----|-----------------|--------------------|
| 1 | Év | Emelt_programozás |
| 2 | 2022május | - |
| 3 | 2022május id.ny | - |
| 4 | 2022október | Virágágyások |
| 5 | 2023május | Ütemezés |
| 6 | 2023május id.ny | - |
| 7 | 2023október | Reklám |
| 8 | 2024május | Beléptető Rendszer |
| 9 | 2024május id.ny | Kráterek |
| 10 | 2024október | Autók mozgása |
| 11 | 2025május | Könyvkiadás |
| 12 | 2025május id.ny | ASCII-rajzok |
| 14 | 2025október | Sebességkorlátozás |



```
1 using namespace std;
2 #include <bits/stdc++.h>
3
4 struct Tadat{
5     int ev, negyedev;
6     string eredet, leiras;
7     int peldany;
```

```
1 EV = 0
2 NEGYEDEV = 1
3 EREDET = 2
4 LEIRAS = 3
5 PLDSZAM = 4
6
7 def feladat1() :
8     kiadasok = list()
9     be = open("kiadas.txt",encoding="utf-8")
10    for sor in be :
11        sor = sor.strip()
12        if sor != "" :
13            sorfeldb = sor.split(';')
14            kiadas = [ int(sorfeldb[EV]), int(sorfeldb[NEGYEDEV]), sorfeldb[EREDET], sorfeldb[LEIRAS], int(sorfeldb[PLDSZAM]) ]
15            kiadasok.append(kiadas)
16    be.close()
17    return kiadasok
18
```

MIRE HASZNÁLHATÓ A MESTERSÉGES INTELLIGENCIA JÓL ÉS ROSSZUL A PROGRAMOZÁSI FELADATOK MEGOLDÁSÁNÁL?

2026. április 22

Siegler Gábor

tanár, Berzsenyi Dániel Gimnázium

A MESTERSÉGES INTELLIGENCIA ESZKÖZEI

AI modellek



ChatGPT

OpenAI fejlesztésű conversational AI



CoPilot

Microsoft AI asszisztens



DeepSeek

Kínai fejlesztésű AI modell



Grok

X (Twitter) AI platformja



Gemini

Google multimodális AI modellje



Claude

Anthropic AI modellje



Meta

Facebook/Meta AI megoldásai



mistral AI

Európai AI

JÓ HASZNÁLAT

- Típus feladatokat jól old meg
- Algoritmust képes készíteni
- Magyaráz
- Lépésekre bont
- Ismeri az „érettségi megoldás” stílust
- Javaslatokat tesz

PROBLÉMÁS HASZNÁLAT

- számonkérés, beadandó – mit is javítunk?, megszokásból nem megy
- elég sokat téved a megoldásokban
- előzzük meg szempontok adásával, promptolással
(adatszerkezet, utasítástípusok, változónevek, mintakimenet,
kerülendő eszközök,...)

ALGORITMUS KÉSZÍTÉS – TARTALMI SZEMPONTOK

1.
Kérjen be két egész számot (a és b) a határok közötti (a határokat is beleértve) p
utasításokat értelmezze!

Minta:

```
Kérem a kisebb számot!  
11  
Kérem a nagyobb számot!  
18  
A páros számok összege:
```

```
Program ParosOsszegzes  
  Változók: a, b, csere, i, osszeg: Egész  
  Be: a, b  
  // 1. Rendezés: Biztosítjuk, hogy 'a' legyen a kisebb  
  Ha a > b Akkor  
    csere := a  
    a := b  
    b := csere  
  Vége ha  
  osszeg := 0  
  // 2. Ciklus a határok között  
  Ciklus i := a-tól b-ig  
    // 3. Pároság vizsgálata  
    Ha i MOD 2 = 0 Akkor  
      osszeg := osszeg + i  
    Vége ha  
  Vége ciklus  
  Ki: osszeg  
Vége program
```

Tervezési kérdések!

Kódolási kérdések a végén!

Egy héten keresztül mértük a reggeli hőmérsékletet. Az adatok ismeretében meg szeretnénk határozni, volt-e reggeli fagy, és ha igen, hány nap.

Az alábbi algoritmus a hőmérsékleti adatok bekérése után ezt határozza meg, de sajnos két helyen azonban hibás. Keresse meg és javítsa ki a hibákat!

Algoritmus:

```
Változók: [ 1]
    n, x, db: Egész [ 2]
Program: [ 3]
    db=0 [ 4]
    Ciklus i:=1-től 10-ig [ 5]
        Ki: "Kérem a hőmérsékletet!" [ 6]
        Be: x [ 7]
        Ha x>0, akkor [ 8]
            db=db+1 [ 9]
        Elágazás vége [10]
    Ciklus vége [11]
    Ha db>0 akkor Ki: "Nem volt fagyos reggel." [12]
    Különben Ki: "A fagyos reggelek száma: ", db [13]
    Elágazás vége [14]
Program vége. [15]
```

Változók:

i, *x*, *db*: Egész [Helyesen *i*-t használunk ciklusváltozónak]

Program:

db := 0

Ciklus *i* := 1-től 7-ig // JAVÍTÁS: Egy héten 7 nap van, nem 10

Ki: "Kérem a hőmérsékletet!"

Be: *x*

Ha $x < 0$ akkor // JAVÍTÁS: Fagy akkor van, ha *x* kisebb, mint 0

db := *db* + 1

Elágazás vége

Ciklus vége

Ha $db > 0$ akkor

Ki: "A fagyos reggelek száma: ", *db* // JAVÍTÁS: Itt írjuk ki a darabszámot

Különben

Ki: "Nem volt fagyos reggel." // JAVÍTÁS: Ha $db=0$, akkor nem volt fagy

Elágazás vége

Program vége.

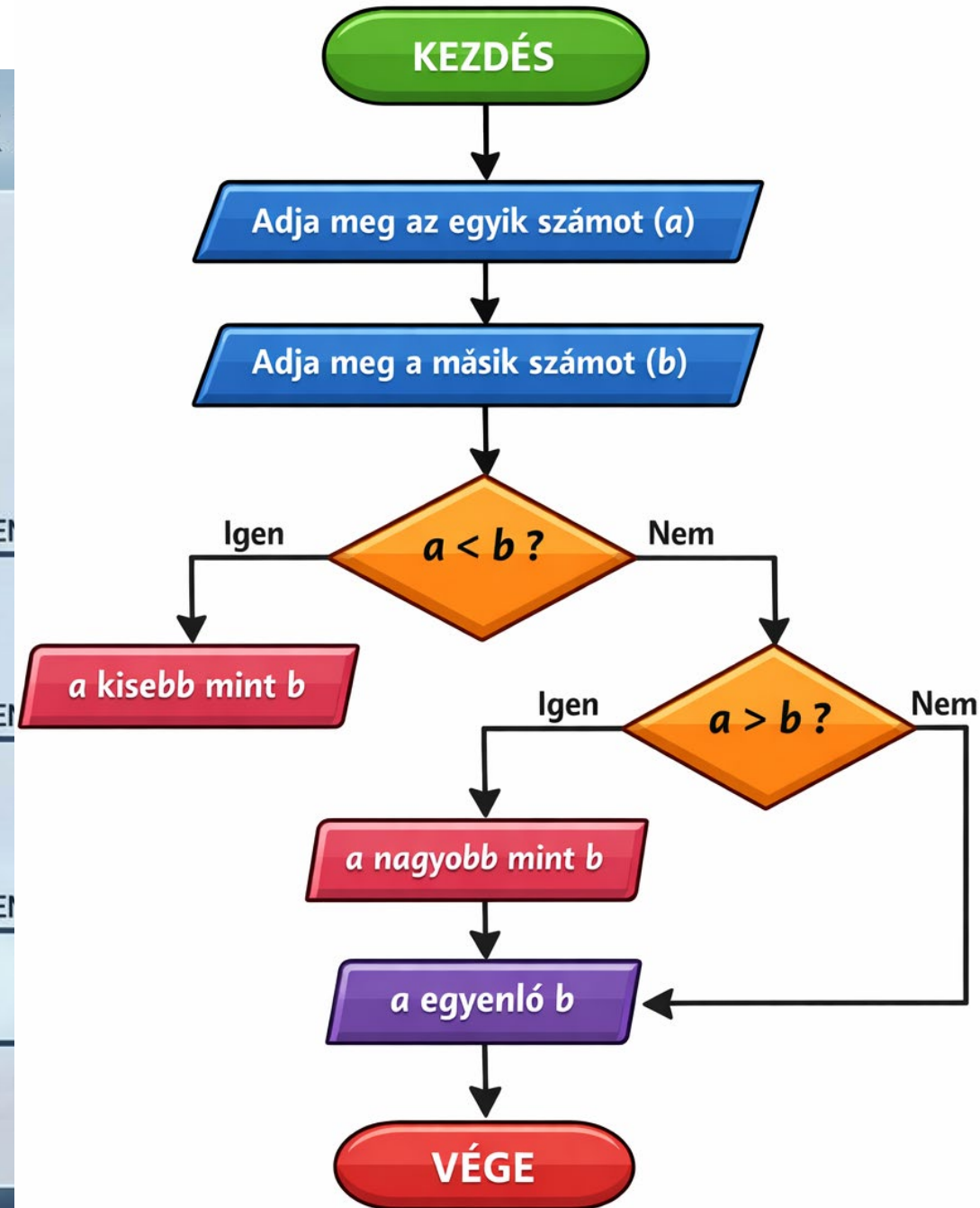
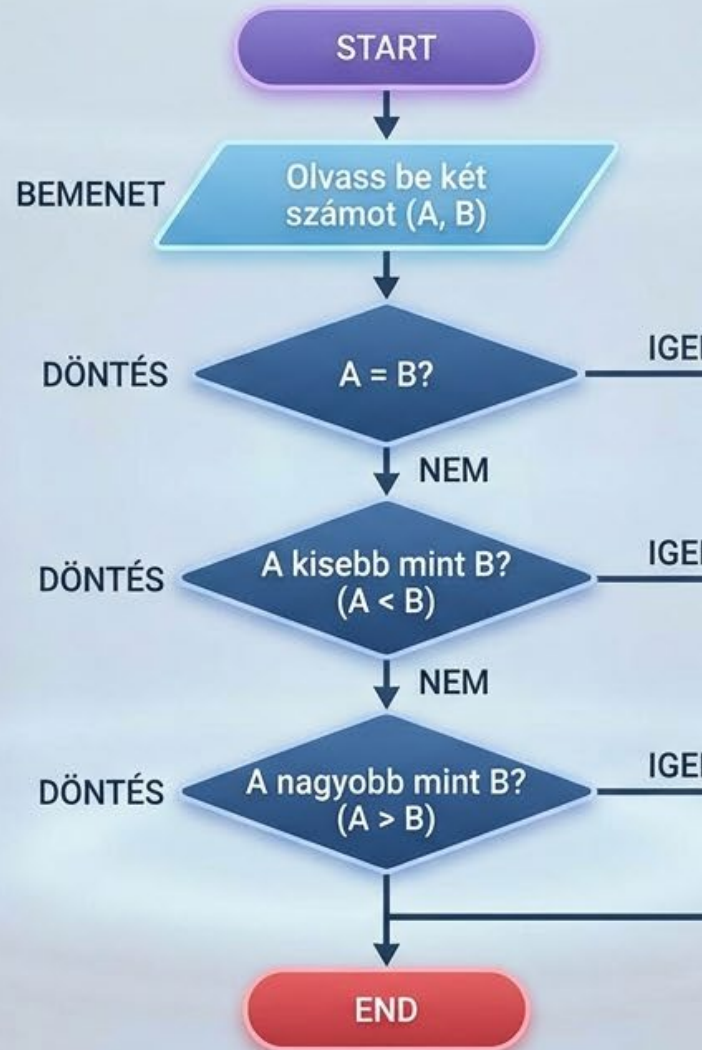
FOLYAMATÁBRA KÉSZÍTÉS

Készítsen algoritmust, amely olvasson be két számot, és írja ki a köztük lévő relációt szöveges formában!

Minta:

```
Adja meg az egyik számot! 12.3  
Adja meg a másik számot! 11.8  
  
11.8 kisebb mint 12.3
```

ALGORITMUS KÉT SZÁM KAPCSOLATÁNAK



NOTEBOOKLM

Példa:

Kérjen be két egész számot (a és b) és készítsen algoritmust, amely összeadja az a és b közötti (a határokat is beleértve) páros számokat! Megoldását, a használt változókat és az utasításokat értelmezze!

Minta:

Kérem a kisebb számot!

11

Kérem a nagyobb számot!

18

A páros számok összege: 60

[Algorithm_Anatomy_NotebookLM.pptx](#)

Problémái: statikus, nem módosítható, nagyon-nagyon lassú, ötletet ad

ROMÁN ÉRETTSÉGI EGY FELADATA

II. TÉTEL (40 pont)

1. Adott a mellékelt algoritmus pszeudokódban.

$a \% b$ jelöli az a természetes számnak a b nem nulla természetes számmal való osztási maradékát.

a. Írja le, hogy mit ír ki az algoritmus a végrehajtása során, ha beolvassuk, ebben a sorrendben, a 20 és 25 számokat. (6p.)

b. Ha az m -be beolvasott szám 5, írjon le két számot az $[1, 50]$ intervallumból, amelyeket az n -be olvashatunk úgy, hogy az algoritmus végrehajtása során, mindkét esetben az utolsó kiírt számérték 2 legyen. (6p.)

c. Írja meg az algoritmusnak megfelelő C/C++ programot. (10p.)

d. Írjon az adott algoritmussal egyenértékű pszeudokód algoritmust, amelyben a minden...végezd el szerkezetet megfelelőképpen helyettesíti egy amíg...végezd el ismétlődő szerkezettel. (6p.)

```
beolvas m, n
      (természetes számok)
ha m < n akkor
  n ← n + m
  m ← n - m
  n ← n - m
■
k ← m
minden i ← m, n, -1 végezd el
  kiír k, ' '
  ha i % 2 = 0 akkor
    k ← k - 1
    kiír '*'
  ■
  k ← k - 1
■
```

TAPASZTALATOK

- Értette a feladatot, az algoritmust
- Képes az algoritmus leírását megváltoztatni, de a „hagyományost” nehéz elérni → a jelöléseket meg kell adni
- Magyaráz: nem egyforma módszerrel → többet érdemes nézni
- Folyamatábra: nehezen, lassan, téveset 😞

FELADATOK KÉSZÍTÉSE

- Egyszerű feladatokat jól készít
- Nem kreatív, érdeemes témát, szereplőket javasolni

Példa:

Készítsél programozási feladatot, amelyben Hófehérke és a 7 törpe szerepel.

Hófehérke kekszeket süt a törpéknek. Véletlenszerűen kiosztja a kekszeket.

Összegzés és maximumkiválasztás tétele legyen a feladatban.

PROGRAMOZÁS ÉS ALGORITMIZÁLÁS A SZÓBELI VIZSGÁN

2026. április 22

Siegler Gábor

tanár, Berzsenyi Dániel Gimnázium

SZÓBELI A VIZSGÁN

- Emelt szinten mindig
 - 30 pont (20%)
 - (120+30) pont \rightarrow 100%
- Középszinten
 - mint a mateknál
 - ha $12\% \leq$ gyakorlati eredmény $< 25\%$
 - ha a gyakorlati $\geq 25\% \Rightarrow$ 100 pont \rightarrow 100%
 - ha van szóbeli \Rightarrow (100 + 50) pont \rightarrow 100%

MIBŐL ÁLL KÖZÉPSZINTEN?

A szóbeli tételsor tartalmi és formai jellemzői

A tételsor jellemzői

A tételsor legfeljebb 20 tételből áll. A tételsor tételeinek legalább 10%-a évenként cserélendő.

Az A) feladatnál felsorolt öt témakörhöz kapcsolódóan legalább két-két tételnek szerepelnie kell a tételsorban.

A tétel jellemzői

A szóbeli tétel két feladatból áll:

- A) feladat: egy téma kifejtése konkrét tartalmi szempontok alapján a Szövegszerkesztés, Számítógépes grafika és képszerkesztés, Bemutatókészítés, Táblázatkezelés, Adatbázis-kezelés témakörökhöz kapcsolódóan. A téma kifejtéséhez számítógépes szemléltetés is kérhető.
- B) feladat: egy programozási feladat megoldása és bemutatása, amelyhez ha a feladat kódolást, kódmódosítást, -értelmezést igényel, akkor internetkapcsolattal nem rendelkező számítógépet használhat a vizsgázó a felkészülési ideje alatt.

A B) feladatot a vizsgázónak a felkészülési időben kell megoldania. A felelet során a vizsgázónak az A) feladatot szóban, önállóan kell kifejtenie és a B) feladathoz elkészített megoldását kell bemutatnia.

A vizsgára megfogalmazott tételek a tételcímeken túl a kifejtéshez konkrét tartalmi szempontokat is felsorolnak.

MIBŐL ÁLL EMELT SZINTEN?

A szóbeli tételsor tartalmi és formai jellemzői

A tételsor jellemzői

Az emelt szintű szóbeli vizsga központi tételsor alapján zajlik. A tételsor legalább 20 tételből áll. A tételsor tételeinek legalább 10%-a évenként cserélendő.

A tétel jellemzői

A szóbeli tétel két tételrészről áll:

- A) feladat: egy téma kifejtése konkrét tartalmi szempontok alapján a Szövegszerkesztés, Számítógépes grafika és képszerkesztés, Bemutatókészítés, Publikálás a világhálón, Táblázatkezelés és Adatbázis-kezelés témakörökhöz kapcsolódóan. Mind a hat témakörhöz kapcsolódó tételnek szerepelnie kell a tételsorban. A téma kifejtéséhez számítógépes szemléltetés is kérhető.
- B) feladat: egy vagy több programozási feladat megoldása és bemutatása. Ha a feladat kódolást, kódmódosítást, -értelmezést igényel, akkor internetkapcsolattal nem rendelkező számítógépet használhat a vizsgázó a felkészülési ideje alatt.

A B) feladatot a vizsgázónak a felkészülési időben kell megoldania. A felelet során a vizsgázónak az A) feladatot szóban, önállóan kell kifejtenie és a B) feladathoz elkészített megoldását kell bemutatnia.

SZÓBELI PONTOZÁS – KÖZÉPSZINT

| | |
|---|----------------|
| Tartalom (Mindent tartalmaz-e, ami a témakörhöz szükséges; vannak-e tárgyi tévedések, rossz magyarázatok.) | 28 pont |
| <i>A) feladat</i> | <i>18 pont</i> |
| <i>B) feladat</i> | <i>10 pont</i> |
| Logikai felépítés (Jó időbeosztás; a lényeg kiemelése; követhető gondolatmenet.): | 8 pont |
| Kifejezőkészség, szaknyelv használata: | 8 pont |
| Kommunikatív készség (Lehet-e a vizsgázót a témában vezetni; ha elakad, megérti-e, amit kérdez a bizottság; lehet-e a vizsgázóval a témáról tartalmas párbeszédet folytatni.): | 6 pont |

SZÓBELI PONTOZÁS – EMELT SZINT

| | |
|--|----------------|
| Tartalom (Mindent tartalmaz-e, ami a témakörhöz szükséges; vannak-e tárgyi tévedések, rossz magyarázatok.) | 18 pont |
| <i>A) feladat</i> | <i>8 pont</i> |
| <i>B) feladat</i> | <i>10 pont</i> |
| Logikai felépítés (Jó időbeosztás; a lényeg kiemelése; követhető gondolatmenet.): | 4 pont |
| Kifejezőkészség, szaknyelv használata: | 4 pont |
| Kommunikatív készség (Lehet-e a vizsgázót a témában vezetni; ha elakad, megérti-e, amit kérdez a bizottság; lehet-e a vizsgázóval a témáról tartalmas párbeszédet folytatni.): | 4 pont |



MINTA TÉTEI KÖZÉPSZINT

B) F Egy tanuló egy adott napi hiányzását egy karaktersorozattal (szöveges adattal) írjuk le, mely X, I és N karaktereket tartalmazhat. Minden karakter egy tanórán való jelenlétét írja le a következőképpen:

- X: a tanuló nem hiányzott
- I: a tanuló igazoltan hiányzott
- N: a tanuló nem igazoltan hiányzott.

Pl.: NXXXII: ez a szöveg azt jelenti, hogy az 1. órától a tanuló igazolatlanul hiányzott, majd három tanórán jelen volt, végül a nap végén az 5. és 6. órától igazoltan hiányzott. Mivel a szöveg 6 karakter hosszú, így aznap 6 tanórája volt.

Készítsen programot Ön által választott programozási nyelven az alábbi feladatok megoldására:

1. Olvasson be egy szöveget, amely jelzi a tanuló adott napi hiányzását. Feltételezheti, hogy csak X, I vagy N karakterekből áll (és szóközt sem tartalmaz).
2. Írja ki, hogy hány órája volt aznap a tanulónak!
3. Írassa ki azoknak az óráknak a sorszámát egymás mellé, szóközzel elválasztva, amelyeken jelen volt a tanuló! Hány ilyen óra volt?

Minta (az ékezetmentes kiírás is megengedett):

```
Kérem a hiányzásokat!  
NIIXXNI  
A tanulónak aznap 7 órája volt.  
A következő órákon volt jelen:  
4 5  
Jelen volt 2 órán.
```

B) feladat

Az alábbi algoritmus egy diák matematika érdemjegyeinek átlagának meghatározására készült, de sajnos két helyen hiba csúszott bele. Keresse meg és javítsa ki a hibákat, majd kódolja az algoritmust az Ön által választott programozási nyelven! (Az ékezetmentes kiírás is megengedett.) Megoldását szóban értelmezze.!

Algoritmus:

```
Változók: [ 1]
    n, ossz, jegy: Egész [ 2]
Program: [ 3]
    Ki: "Érdemjegyek száma?" [ 4]
    Be: n [ 5]
    ossz:=1 [ 6]
    Ciklus i:=1-től n-ig [ 7]
        Be: jegy [ 8]
        ossz=ossz+jegy [ 9]
    Ciklus vége [10]
    Ki: "A jegyek átlaga:", ossz/10 [11]
Program vége. [12]
```

MINTA TÉTEL EMELT SZINT

B) FELADAT

Készítsen egy algoritmust az Ön által ismert egyik algoritmusleíró eszköz segítségével az alábbi feladathoz, és azt mutassa be!

A téli időszakban N nap rögzítettük a hőmérsékletet. Fagyos időszaknak nevezzük az egymás utáni olyan napok sorozatát, amikor a hőmérséklet 0 fok alatt volt, de az időszakot közvetlenül megelőző és követő nap pedig nemnegatív volt.

Az adatokat az *Egy lehetséges algoritmus:*

hőmérsékletet az

Az algoritmus 1
rögzített sorozat
rögzített érték, v
rögzített nemneg

Az algoritmusál
(listában), azok 1

Például:



Bemene

Változók:

db: Egész

H: Tömb (1..N: Egész): Valós

Fagyos időszakok:

db := 0

Ha $H[1] < 0$ akkor db := 1 különben db := 0

Ciklus $i=2$ -től N -ig

Ha $(H[i-1] \geq 0)$ és $(H[i] < 0)$ akkor db := db + 1

Ciklus vége

Ki: db

Eljárás vége

Köszönöm a figyelmet!

Siegler Gábor

siegler.gabor@gmail.com