

Kódjavítás (14 pont)

Az alábbi PlanG kódot javítsd, illetve egészítsd ki úgy, hogy a megadott specifikációnak eleget tegyen:

a) (6 pont)

bemenet: valós számok 10 hosszú (s) sorozata

kimenet: a -10 és 10 közötti nem nulla számok száma (c)

```
PROGRAM intervallum
  VÁLTOZÓK:
    s : VALÓS[10],
    i, c : EGÉSZ

  i := 1
  CIKLUS AMÍG i < 10
    BE : s[i]
  CIKLUS_VÉGE
  CIKLUS
    i := i+1
    HA ( | s[i] | > 0 ) AKKOR c = 1
    AMÍG i < 10
  HA_VÉGE
  KI: c
PROGRAM_VÉGE
```

b) (8 pont)

bemenet: egy 10 elemű szöveg sorozat (szavak) egy bemenő fájlban (bemenet)

kimenet: az 'a' betűvel kezdődő szavak, majd a 'b' betűvel kezdődő szavak a kimeneten

```
PROGRAM a_b_betűs_szavak
  VÁLTOZÓK:
    bemenet : FÁJL
    szavak: SZÖVEG[10],
    i: EGÉSZ[10]
  i := 0
  CIKLUS AMÍG (i<10)
    BE bemenet : szavak[i]
    HA (szavak[i][0] = 'a') AKKOR
      KI: szavak
    HA_VÉGE
    i := i + 1
    LEZÁR bemenet
  CIKLUS_VÉGE
  i := 0
  CIKLUS AMÍG (i<20)
    HA (szavak[i][0] = 'b') AKKOR
      KI bemenet : szavak[i]
  CIKLUS_VÉGE
PROGRAM_VÉGE
```

. Kódjavítás (14 pont)

Az alábbi PlanG kódot javítsd, illetve egészítsd ki úgy, hogy a megadott specifikációnak eleget tegyen:

a) (6 pont)

bemenet: egész számok (a) sorozata, amit 0 zár

kimenet: a számok szorzata

```
PROGRAM 0igszorzas
  VÁLTOZÓK:
    a: EGÉSZ,
    s: KARAKTER
  s := 10
  CIKLUS
    s := s + a
  BE: a
  AMÍG (a /= 0)
  KI: a
PROGRAM_VÉGE
```

b) (8 pont)

bemenet: egy 10 elemű szöveg tömb (szavak)

kimenet: az 'a' betűvel kezdődő szavak, valamint a 'b' betűvel kezdődő szavak

```
PROGRAM a_b_betűs_szavak
  VÁLTOZÓK:
    bemenet : FÁJL
    szavak: SZÖVEG[10],
    i: EGÉSZ[10]
  i := 0
  CIKLUS AMÍG (i<10)
    BE bemenet : szavak[i]
    HA (szavak[i][0] = 'a') AKKOR
      KI: szavak
    HA_VÉGE
    i := i + 1
  LEZÁR bemenet
  CIKLUS_VÉGE
  i := 0
  CIKLUS AMÍG (i<20)
    HA (szavak[i][0] = 'b') AKKOR
      KI bemenet : szavak[i]
    CIKLUS_VÉGE
PROGRAM_VÉGE
```

Kódjavítás:

Javítsd ki az alábbi programot úgy, hogy megfeleljen az alábbi specifikációnak:

Bemenet: egy fájlban (f) tárolt egész számok (a) sorozata.

Kimenet: a páratlan számok közül a legnagyobb.

```
PROGRAM legnagyobb_páratlan
  VÁLTOZÓK:
    a, max: VALÓS,
    s: EGÉSZ,
    f: BEFÁJL
  a := 0
  f: MEGNYIT
  max := a
  CIKLUS
```

```

    BE: a
    HA (NEM VÉGE f)
    AKKOR
        max := a
    AMÍG a
    CIKLUS_VÉGE
    KI: „A legnagyobb páratlan szám: ”, a
    LEZÁR f
PROGRAM_VÉGE

```

Kódjavítás (15 pont)

Az alábbi PlanG kódot javítsd, illetve egészítsd ki úgy, hogy a megadott feladatot végezze:

Keresd meg egy 10 hosszú valós sorozatban hány pozitív szám van.

```

PROGRAM intervallum
VÁLTOZÓK:
    s : EGÉSZ[10], i, e : EGÉSZ

    i := 1
    CIKLUS AMÍG i < 10
        BE : s[a]
    CIKLUS_VÉGE
    CIKLUS
        i := i+1
        HA NEM ( | s[i] | > 0 ) AKKOR e = 1
        AMÍG i < 10
    HA_VÉGE
    KI: e
PROGRAM_VÉGE

```

Kódjavítás (15 pont)

Az alábbi PlanG kódot javítsd, illetve egészítsd ki úgy, hogy a megadott feladatot végezze:

Keresd meg egy több soros mondatban a szavak számát. (A mondat ponttal végződik.)

```

PROGRAM mondat
VÁLTOZÓK:
    mond : KARAKTER[100], i : EGÉSZ, o : LOGIKAI

    i := 0
    o := 0
    CIKLUS AMÍG (mond[i] = ".")
        KI : mond[i]
        i := i+1
    AMÍG (mond[i] = ".")
    CIKLUS AMÍG (i>0)
        HA ( mond[i] @ " " ) AKKOR o = o + i;
        HA_VÉGE
    CIKLUS_VÉGE
    KI: mond
PROGRAM_VÉGE

```