TÁRSAS

2023. október emelt informatika érettségi
programozási feladat

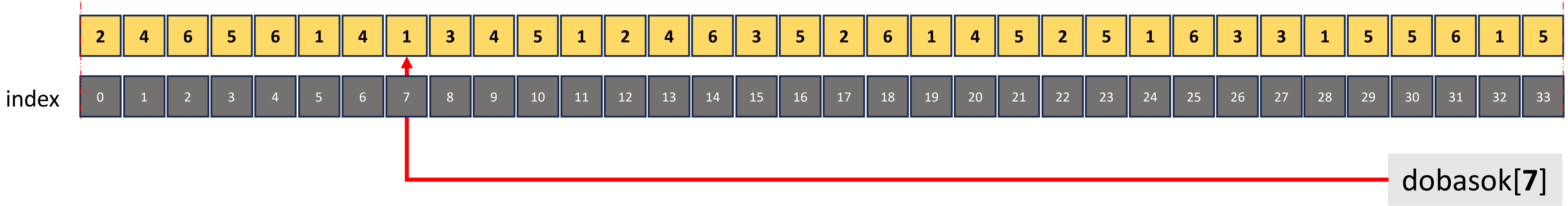
1. ADATOK BEOLVASÁSA

osvenyek (lista) ⇐ osvenyek.txt



lista	elemek
osvenyek	M, V és E karakterekből álló változó hosszúságú szövegek
dobasok	1 és 6 közötti egész számok

dobasok (lista) ⇐ dobasok.txt

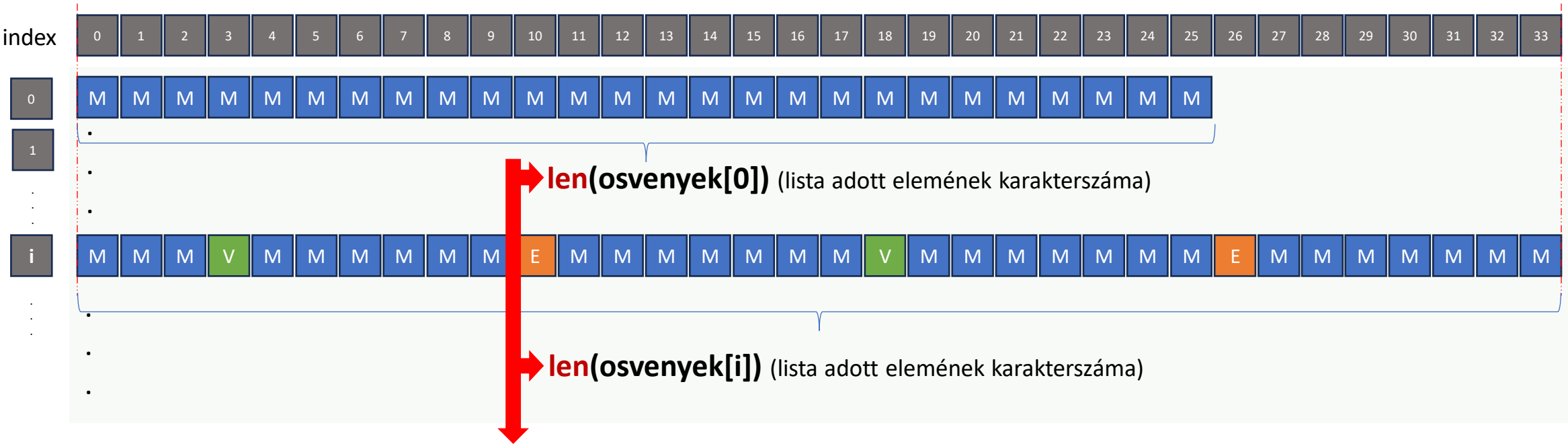


2. DOBÁSOK ÉS ÖSVÉNYEK SZÁMA



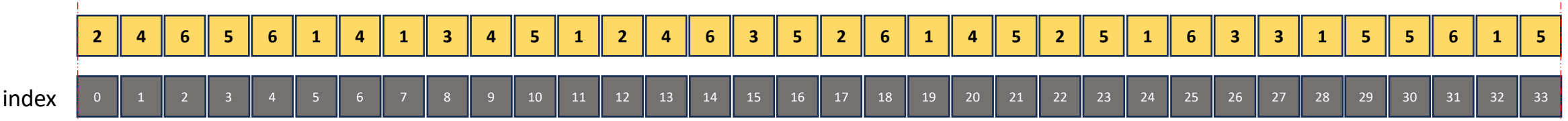
3. LEGHOSSZABB ÖSVÉNY ADATAI

osvenyek (lista)

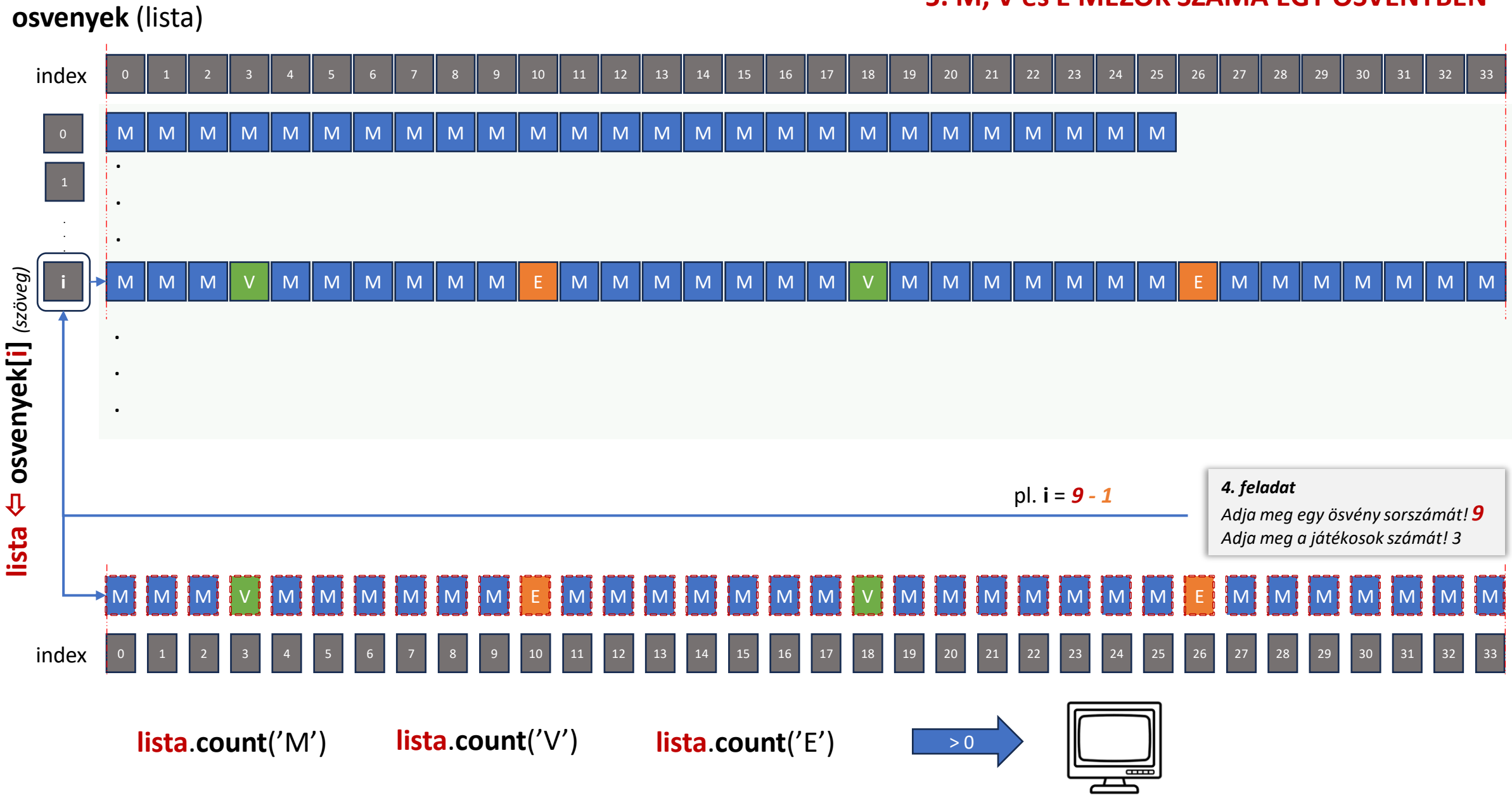


Ezek közül a maximális!
(MAX, index+1)

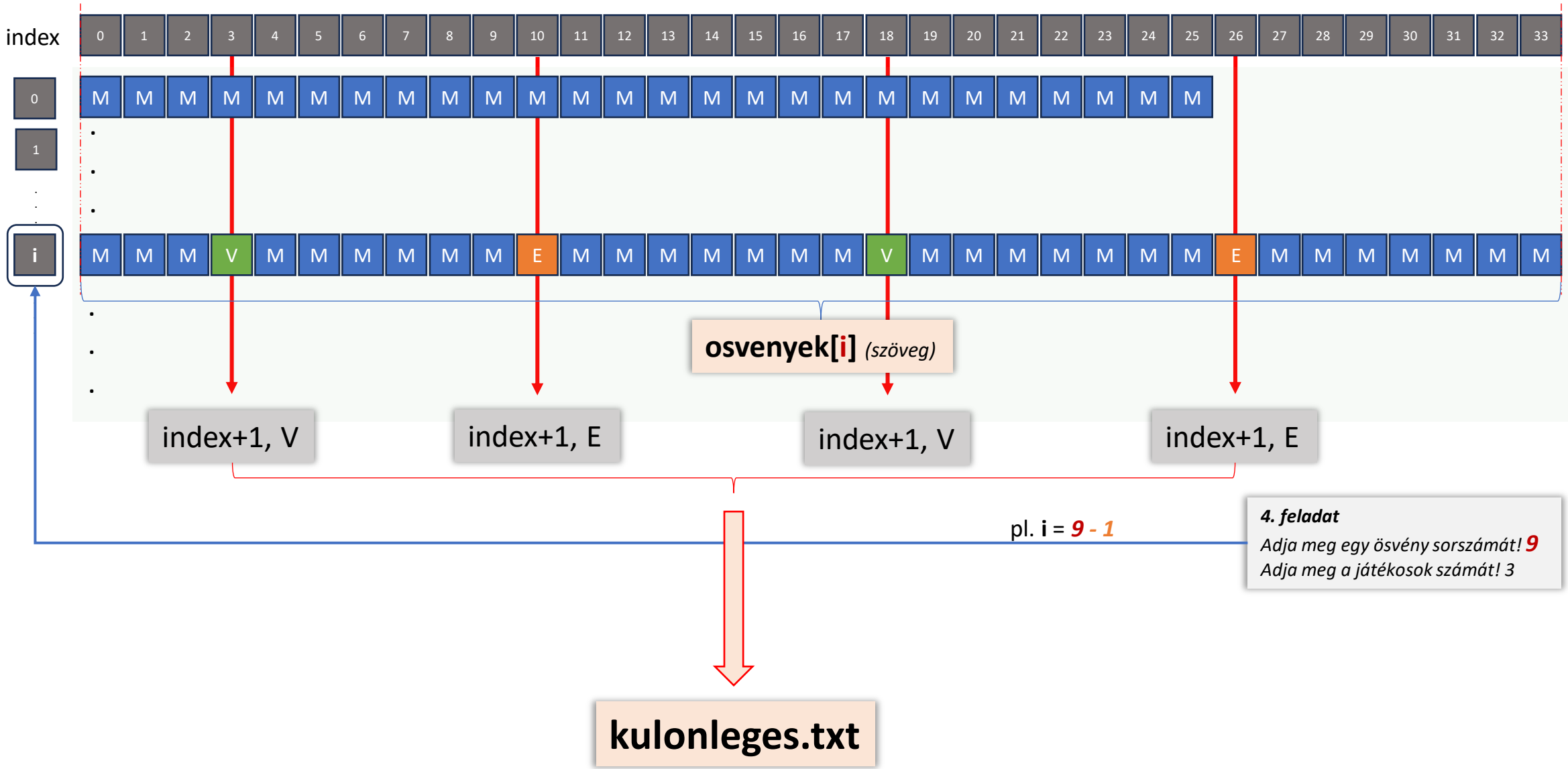
dobasok (lista)

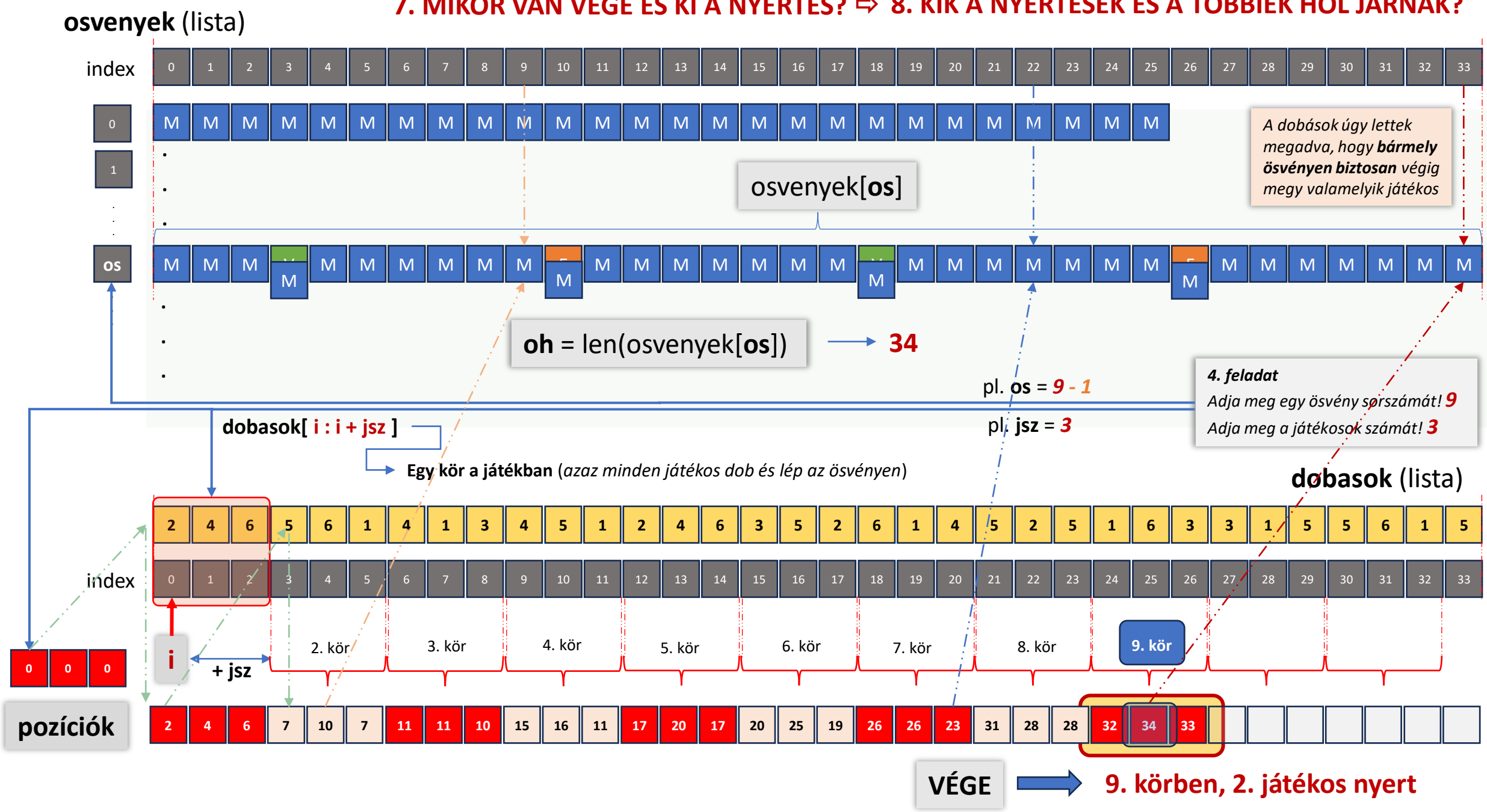


5. M, V és E MEZŐK SZÁMA EGY ÖSVÉNYBEN



6. KÜLÖNLEGES MEZŐK ADATAINAK FÁJLBA ÍRÁSA




7. MIKOR VAN VÉGE ÉS KI A NYERTES? ⇨ 8. KIK A NYERTESEK ÉS A TÖBBIEK HOL JÁRNAK?
osvenyek (lista)
index
os
osvenyek[os]
A dobások úgy lettek megadva, hogy bármely ösvényen biztosan végig megy valamelyik játékos
oh = len(osvenyek[os])
34
pl. os = 9 - 1
pl. jsz = 3
4. feladat
Adja meg egy ösvény sorszámát! 9
Adja meg a játékosok számát! 3
dobasok[i : i + jsz]
Egy kör a játékban (azaz minden játékos dob és lép az ösvényen)
dobasok (lista)
i
+ jsz
2. kör
3. kör
4. kör
5. kör
6. kör
7. kör
8. kör
9. kör
pozíciók
VÉGE
9. körben, 2. játékos nyert

KÓDOK

PYTHON kódrészletek

„normál” kódolás

```
def F1():
    global osvenyek, dobasok
    osvenyek=[]
    fbe=open('osvenyek.txt','r')
    for sor in fbe:
        sor=sor.strip()
        osvenyek.append(sor)
    fbe.close()

    fbe=open('dobasok.txt','r')
    sor=fbe.readline().strip().split()
    dobasok=list(map(int,sor))
    fbe.close()
```


def F1():
    global osvenyek, dobasok
    osvenyek=[]
    fbe=open('osvenyek.txt','r')
    for sor in fbe:
        sor=sor.strip()
        osvenyek.append(sor)
    fbe.close()

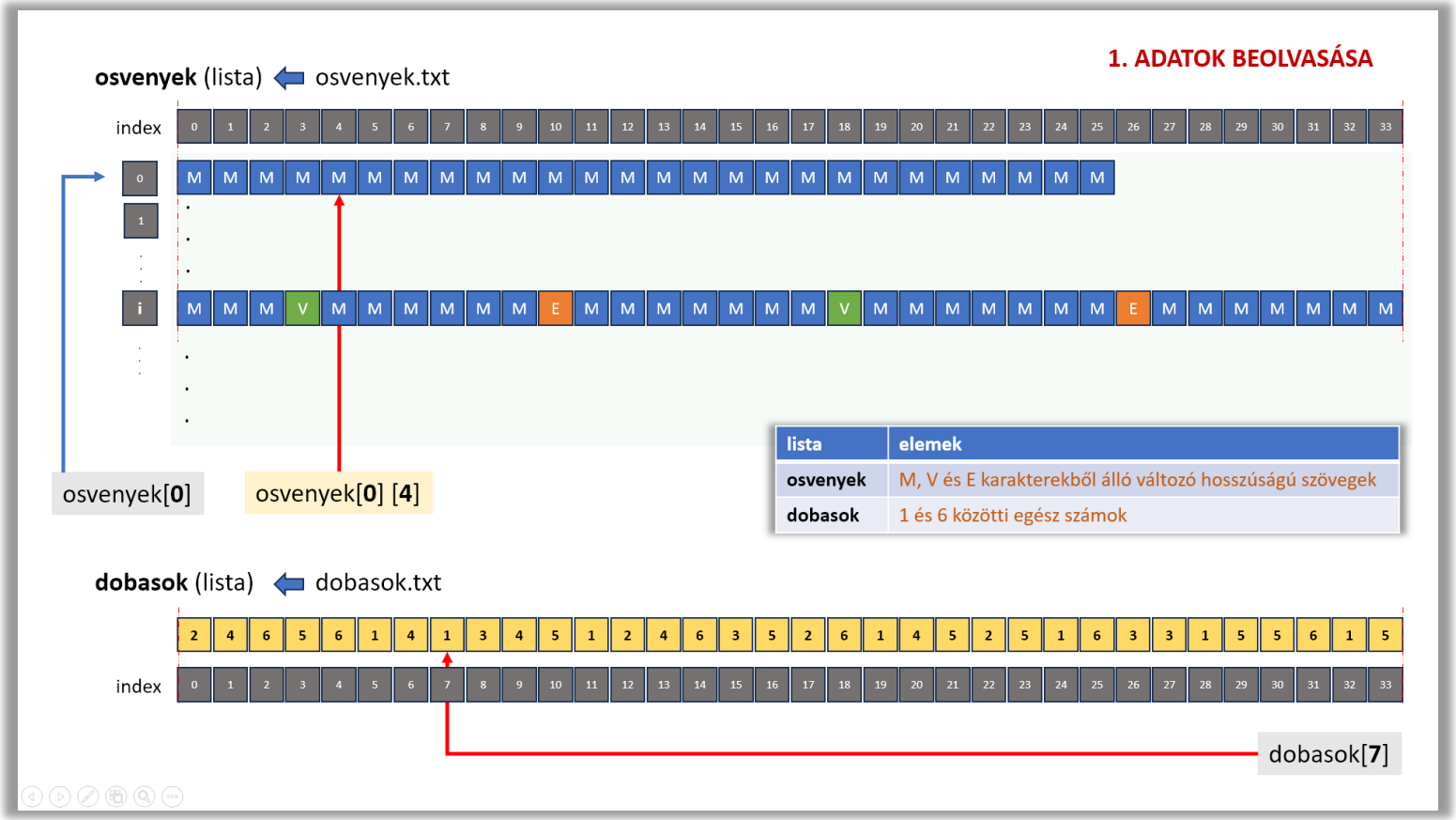
    fbe=open('dobasok.txt','r')
    sor=fbe.readline().strip().split()
    dobasok=list(map(int,sor))
    fbe.close()

1. ADATOK BEOLVASÁSA

„rövid” kódolás

```
osvenyek=[sor.strip() for sor in open('osvenyek.txt','r')]
dobasok=list(map(int,open('dobasok.txt','r').readline().strip().split()))
```


osvenyek=[sor.strip() for sor in open('osvenyek.txt','r')]
dobasok=list(map(int,open('dobasok.txt','r').readline().strip().split()))



„normál” kódolás

2. DOBÁSOK ÉS ÖSVÉNYEK SZÁMA

```
15 def F2():
16     print('2. feladat')
17     print(f'A dobások száma: {len(dobasok)}')
18     print(f'Az ösvények száma: {len(osvenyek)}')
```

„rövid” kódolás

```
6 feladat(2)
7 print(f'A dobások száma: {len(dobasok)}')
8 print(f'Az ösvények száma: {len(osvenyek)}')
```

```
def F2():
    print('2. feladat')
    print(f'A dobások száma: {len(dobasok)}')
    print(f'Az ösvények száma: {len(osvenyek)}')
```

```
feladat(2)
print(f'A dobások száma: {len(dobasok)}')
print(f'Az ösvények száma: {len(osvenyek)}')
```

```
def feladat(f):
    print(f'{'-'*75}\n{f}. feladat')
```



3. LEGHOSSZABB ÖSVÉNY ADATAI

„normál” kódolás

```
20 def F3():
21     print('3. feladat')
22     # PROGRAMOZÁSI TÉTEL - szélsőértékkeresés (max,min)
23     maxh=len(osvenyek[0])
24     ssz=1
25     for i in range(1,len(osvenyek)):
26         if len(osvenyek[i])>maxh:
27             maxh=len(osvenyek[i])
28             ssz=i+1
29     print(f'Az egyik leghosszabb a(z) {ssz}. ösvény, hossza: {maxh}')
30
31     rendezett=sorted(osvenyek,key=lambda elem:len(elem),reverse=True)
32     maxh=len(rendezett[0])
33     ssz=osvenyek.index(rendezett[0])+1
34     # legn=max(osvenyek,key=lambda elem:len(elem))
35     print(f'Az egyik leghosszabb a(z) {ssz}. ösvény, hossza: {maxh}')
36
```

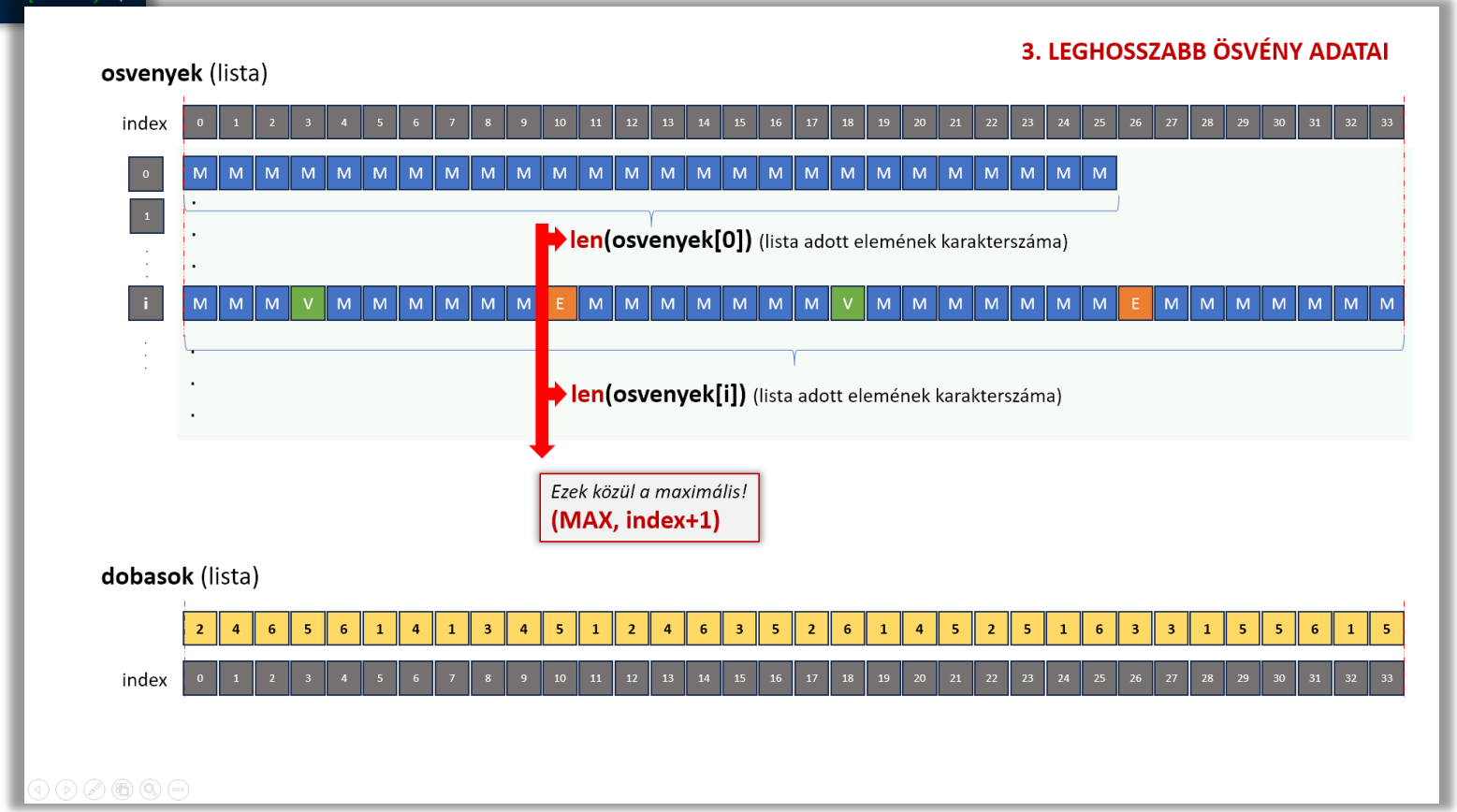
```
def F3():
    print('3. feladat')
    # PROGRAMOZÁSI TÉTEL - szélsőértékkeresés (max,min)
    maxh=len(osvenyek[0])
    ssz=1
    for i in range(1,len(osvenyek)):
        if len(osvenyek[i])>maxh:
            maxh=len(osvenyek[i])
            ssz=i+1
    print(f'Az egyik leghosszabb a(z) {ssz}. ösvény, hossza: {maxh}')

    rendezett=sorted(osvenyek,key=lambda elem:len(elem),reverse=True)
    maxh=len(rendezett[0])
    ssz=osvenyek.index(rendezett[0])+1
    # legn=max(osvenyek,key=lambda elem:len(elem))
    print(f'Az egyik leghosszabb a(z) {ssz}. ösvény, hossza: {maxh}')
```

„rövid” kódolás

```
10 feladat(3)
11 maxh_osv=max(osvenyek,key=lambda osveny:len(osveny))
12 ssz=osvenyek.index(maxh_osv)+1
13 print(f'Az egyik leghosszabb a(z) {ssz}. ösvény, hossza: {len(maxh_osv)}')
14
```

```
feladat(3)
maxh_osv=max(osvenyek,key=lambda osveny:len(osveny))
ssz=osvenyek.index(maxh_osv)+1
print(f'Az egyik leghosszabb a(z) {ssz}. ösvény, hossza: {len(maxh_osv)}')
```



„normál” kódolás

```
def F4():
    print('4. feladat')
    global os,jsz
    os=int(input('Adja meg egy ösvény sorszámát! '))
    jsz=int(input('Adja meg a játékosok számát! '))

def F5():
    print('5. feladat')
    osveny=osvenyek[os-1]
    lista=list(osveny)
    M=lista.count('M')
    V=lista.count('V')
    E=lista.count('E')
    if M>0:
        print(f'M: {M} darab')
    if V>0:
        print(f'V: {V} darab')
    if E>0:
        print(f'E: {E} darab')
```

```
def F4():
    print('4. feladat')
    global os,jsz
    os=int(input('Adja meg egy ösvény sorszámát! '))
    jsz=int(input('Adja meg a játékosok számát! '))

def F5():
    print('5. feladat')
    osveny=osvenyek[os-1]
    lista=list(osveny)
    M=lista.count('M')
    V=lista.count('V')
    E=lista.count('E')
    if M>0:
        print(f'M: {M} darab')
    if V>0:
        print(f'V: {V} darab')
    if E>0:
        print(f'E: {E} darab')
```

„rövid” kódolás

5. M, V és E MEZŐK SZÁMA EGY ÖSVÉNYBEN

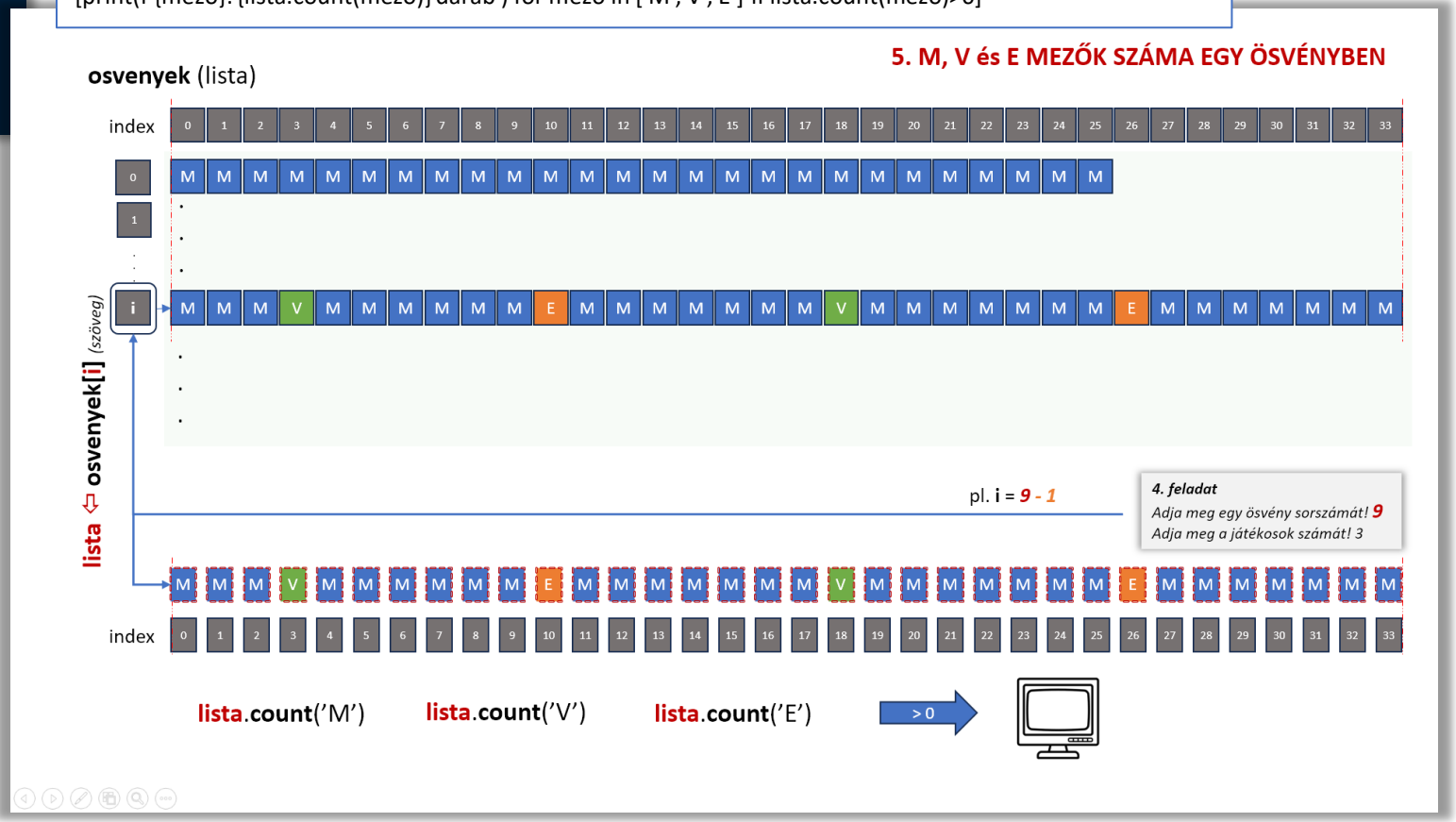
```
feladat(4)
osveny=osvenyek[int(input('Adja meg egy ösvény sorszámát! '))-1]
jsz=int(input('Adja meg a játékosok számát! '))

feladat(5)
lista=list(osveny)
[print(f'{mezo}: {lista.count(mezo)} darab') for mezo in ['M','V','E'] if lista.count(mezo)>0]
```

```
feladat(4)
osveny=osvenyek[int(input('Adja meg egy ösvény sorszámát! '))-1]
jsz=int(input('Adja meg a játékosok számát! '))

feladat(5)
lista=list(osveny)
[print(f'{mezo}: {lista.count(mezo)} darab') for mezo in ['M','V','E'] if lista.count(mezo)>0]
```

5. M, V és E MEZŐK SZÁMA EGY ÖSVÉNYBEN



6. KÜLÖNLEGES MEZŐK ADATAINAK FÁJLBA ÍRÁSA

„normál” kódolás

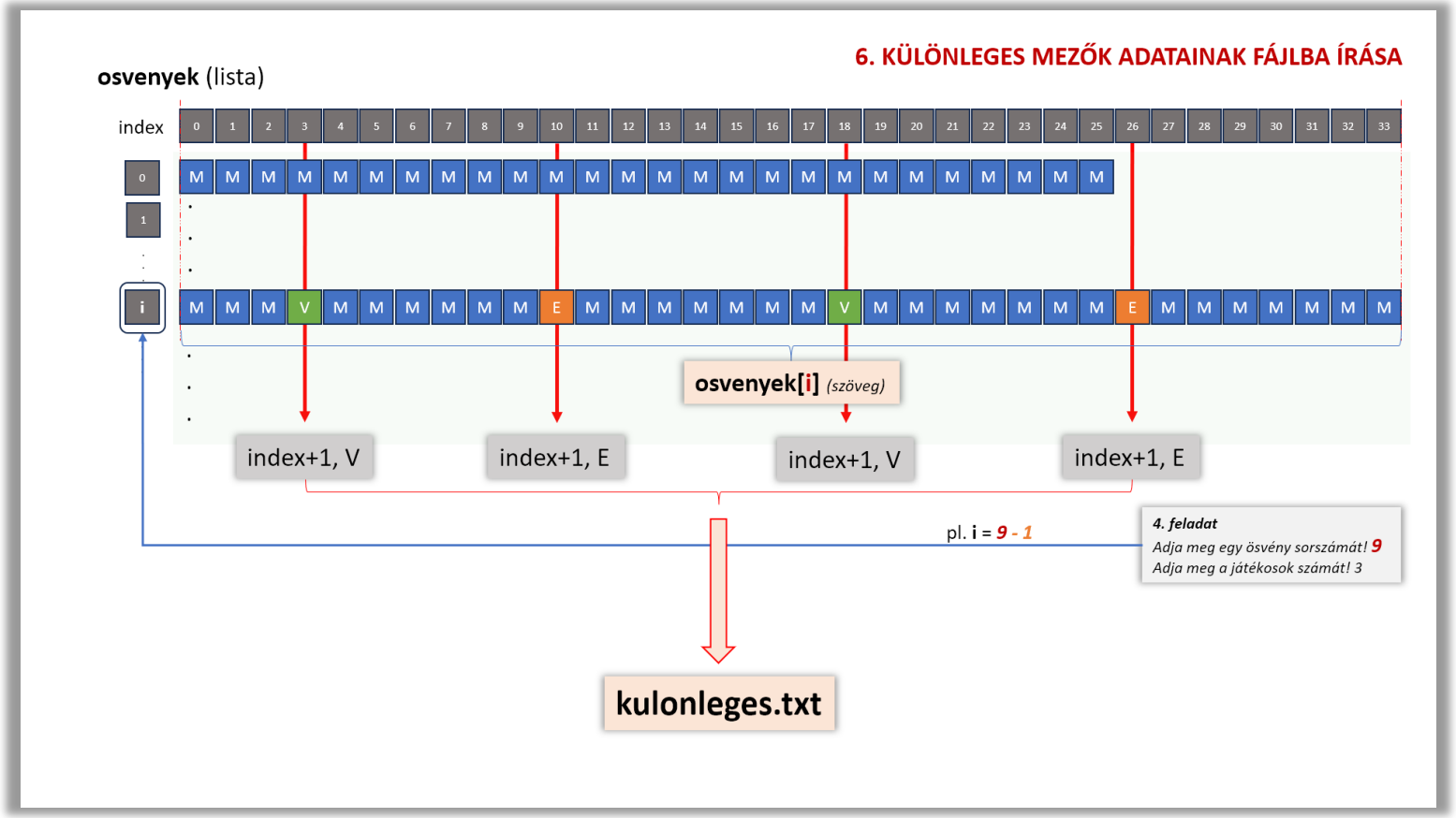
```
57 def F6():
58     fki=open('kulonleges.txt','w')
59     osveny=osvenyek[os-1]
60     for i in range(len(osveny)):
61         if osveny[i]!='M':
62             print(f'{i+1}\t{osveny[i]}',file=fki)
63     fki.close()
```

```
def F6():
    fki=open('kulonleges.txt','w')
    osveny=osvenyek[os-1]
    for i in range(len(osveny)):
        if osveny[i]!='M':
            print(f'{i+1}\t{osveny[i]}',file=fki)
    fki.close()
```

„rövid” kódolás

```
24 with open('kulonleges.txt','w') as fki:
25     [print(f'{i+1}\t{osveny[i]}',file=fki)for i in range(len(osveny)) if osveny[i]!='M']
```

```
with open('kulonleges.txt','w') as fki:
    [print(f'{i+1}\t{osveny[i]}',file=fki) for i in range(len(osveny)) if osveny[i]!='M']
```



„normál” kódolás

„rövid” kódolás

7. MIKOR VAN VÉGE ÉS KI A NYERTES?

```
def F7():
    print('7. feladat')
    kor=0
    ukor=0
    oh=len(osvenyek[os-1])
    szum=[0]*jsz
    for i in range(0,len(dobasok)-jsz,jsz):
        adott_kor=dobasok[i:i+jsz]
        for j in range(jsz):
            szum[j]+=adott_kor[j]
        kor+=1
        for j in range(jsz):
            if szum[j]>=oh:
                ukor=kor
                jsorszam=j+1
        if ukor!=0:
            break

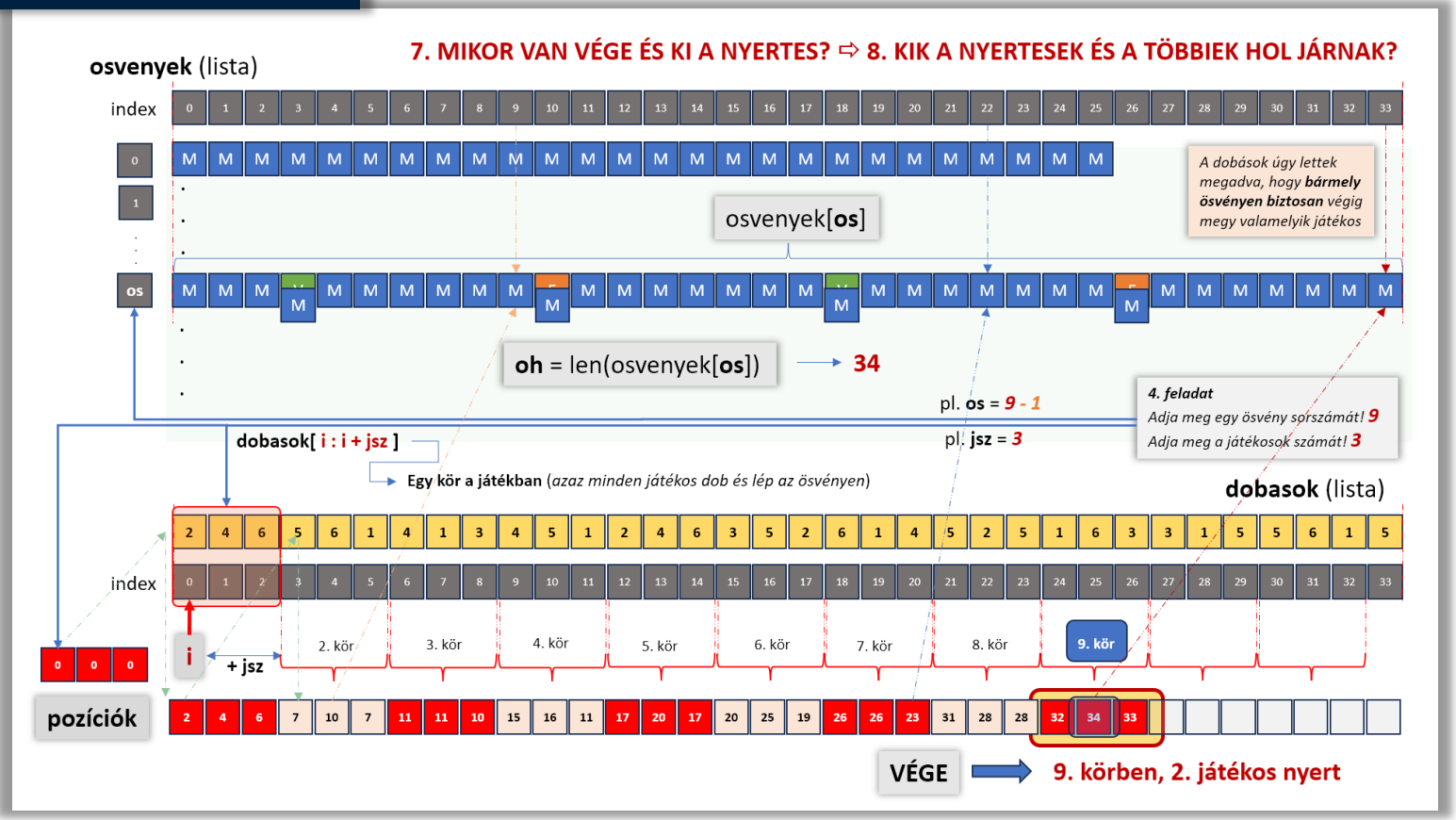
    print(f'A játék a(z) {ukor}. körben fejeződött be. A legtávolabb jutó(k) sorszáma: {jsorszam}')
```

```
feladat(7)
poziciok=[0]*jsz
for i in range(0,len(dobasok)-jsz,jsz):
    aktualis_kor=dobasok[i:i+jsz]
    for j in range(jsz):
        poziciok[j]+=aktualis_kor[j]
    if max(poziciok)>=len(osveny):
        nyertes=poziciok.index(max(poziciok))+1
        break
print(f'A játék a(z) {i//jsz+1}. körben fejeződött be. \
A legtávolabb jutó(k) sorszáma: {nyertes}')
```

```
feladat(7)
poziciok=[0]*jsz
for i in range(0,len(dobasok)-jsz,jsz):
   aktualis_kor=dobasok[i:i+jsz]
   for j in range(jsz):
      poziciok[j]+=aktualis_kor[j]
   if max(poziciok)>=len(osveny):
      nyertes=poziciok.index(max(poziciok))+1
      break
print(f'A játék a(z) {i//jsz+1}. körben fejeződött be. \
A legtávolabb jutó(k) sorszáma: {nyertes}')
```

```
def F7():
   print('7. feladat')
   kor=0
   ukor=0
   oh=len(osvenyek[os-1])
   szum=[0]*jsz
   for i in range(0,len(dobasok)-jsz,jsz):
      adott_kor=dobasok[i:i+jsz]
      for j in range(jsz):
         szum[j]+=adott_kor[j]
      kor+=1
      for j in range(jsz):
         if szum[j]>=oh:
            ukor=kor
            jsorszam=j+1
      if ukor!=0:
         break

   print(f'A játék a(z) {ukor}. körben fejeződött be. \
   A legtávolabb jutó(k) sorszáma: {jsorszam}')
```



„normál” kódolás

```python
def F8():
    print('8. feladat')
    kor=0
    ukor=0
    oh=len(osvenyek[os-1])
    szum=[0]*jsz
    nyertesek=[]
    for i in range(0,len(dobasok)-jsz,jsz):
        adott_kor=dobasok[i:i+jsz]
        for j in range(jsz):
            hozzaad=adott_kor[j]
            szum[j]+=hozzaad

            if szum[j]<oh:
                holvan=osvenyek[os-1][szum[j]-1]
                if holvan=='E':
                    szum[j]+=hozzaad
                if holvan=='V':
                    szum[j]-=hozzaad
        kor+=1
        for j in range(jsz):
            if szum[j]>=oh:
                ukor=kor
                nyertesek.append(j+1)
        if ukor!=0:
            break

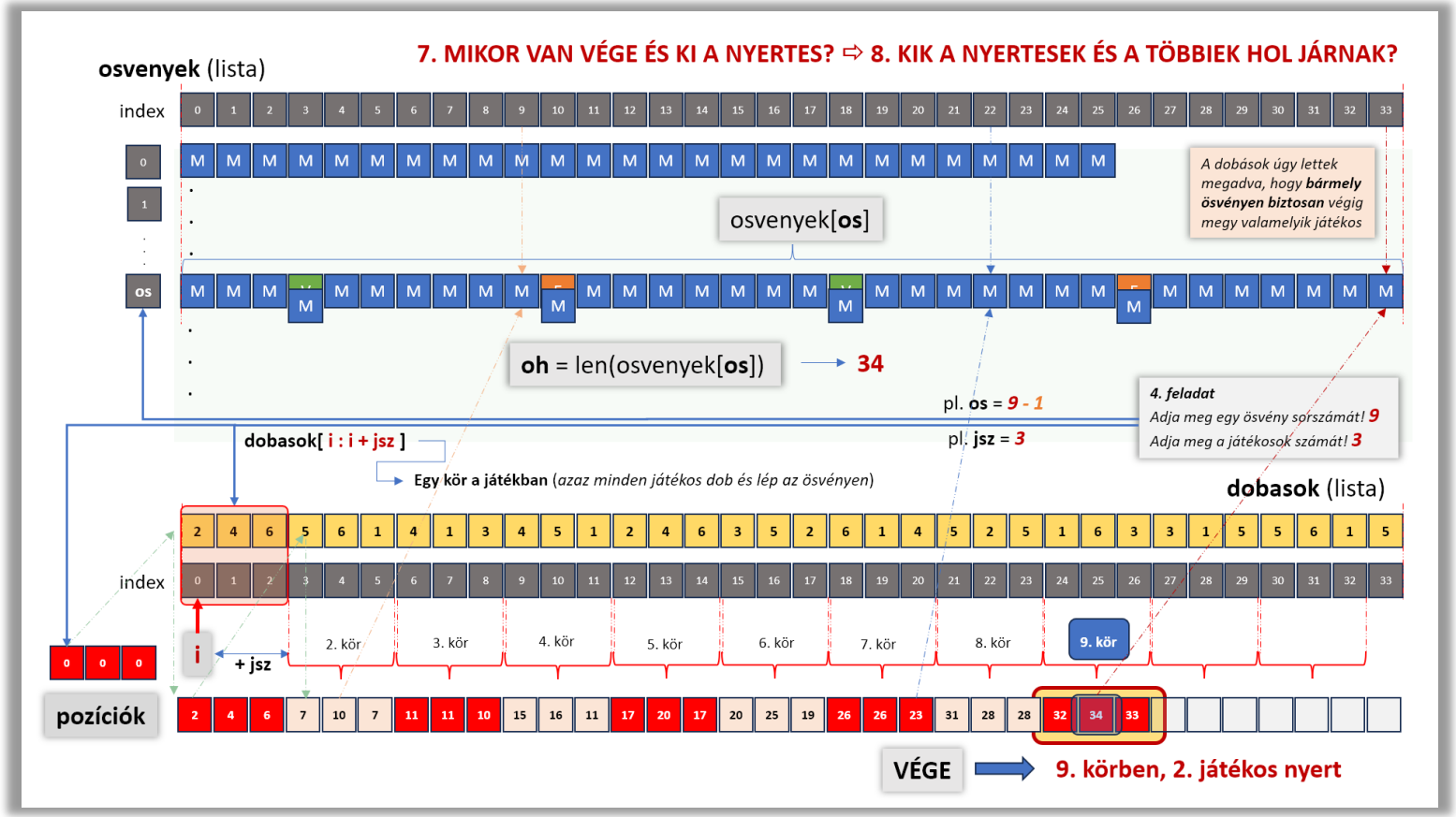
    print('Nyertesek:',end=' ')
    for i in range(len(nyertesek)):
        print(nyertesek[i],end=' ')
    print('\nA többiek pozíciója:')
    for i in range(jsz):
        if i+1 not in nyertesek:
            print(f'{i+1}. játékos, {szum[i]}. mező')
```

„rövid” kódolás

```python
feladat(8)
poziciok=[0]*jsz
nyertesek=[]
for i in range(0,len(dobasok)-jsz,jsz):
    aktualis_kor=dobasok[i:i+jsz]
    for j in range(jsz):
        poziciok[j]+=aktualis_kor[j]
        if poziciok[j]<len(osveny):
            match osveny[poziciok[j]-1]:
                case 'E': poziciok[j]+=aktualis_kor[j]
                case 'V': poziciok[j]-=aktualis_kor[j]

    if len([nyertesek.append(j+1) for j in range(jsz) if poziciok[j]>=len(osveny)])>0:
        break
print('Nyertesek:',end=' ')
[print(nyertesek[i],end=' ') for i in range(len(nyertesek))]
print(f'\nA többiek pozíciója a(z) {len(osveny)} hosszú ösvényen:')
[print(f'{i+1}. játékos: {poziciok[i]}. mező')for i in range(jsz) if i+1 not in nyertesek]
```

8. KIK A NYERTESEK ÉS A TÖBBIEK HOL JÁRNAK?



8. KIK A NYERTESEK ÉS A TÖBBIEK HOL JÁRNAK?

„normál” kódolás

```
def F8():
    print('8. feladat')
    kor=0
    ukor=0
    oh=len(osvenyek[os-1])
    szum=[0]*jsz
    nyertesek=[]
    for i in range(0,len(dobasok)-jsz,jsz):
        adott_kor=dobasok[i:i+jsz]
        for j in range(jsz):
            hozzaad=adott_kor[j]
            szum[j]+=hozzaad

            if szum[j]<oh:
                holvan=osvenyek[os-1][szum[j]-1]
                if holvan=='E':
                    szum[j]+=hozzaad
                if holvan=='V':
                    szum[j]-=hozzaad
        kor+=1
        for j in range(jsz):
            if szum[j]>=oh:
                ukor=kor
                nyertesek.append(j+1)
        if ukor!=0:
            break

    print('Nyertesek:',end=' ')
    for i in range(len(nyertesek)):
        print(nyertesek[i],end=' ')
    print('\nA többiek pozíciója:')
    for i in range(jsz):
        if i+1 not in nyertesek:
            print(f'{i+1}. játékos, {szum[i]}. mező')
```

„rövid” kódolás

```
feladat(8)
poziciok=[0]*jsz
nyertesek=[]
for i in range(0,len(dobasok)-jsz,jsz):
    aktualis_kor=dobasok[i:i+jsz]
    for j in range(jsz):
        poziciok[j]+=aktualis_kor[j]
        if poziciok[j]<len(osveny):
            match osveny[poziciok[j]-1]:
                case 'E': poziciok[j]+=aktualis_kor[j]
                case 'V': poziciok[j]-=aktualis_kor[j]
    if len([nyertesek.append(j+1) for j in range(jsz) if poziciok[j]>=len(osveny)])>0:
        break
print('Nyertesek:',end=' ')
[print(nyertesek[i],end=' ') for i in range(len(nyertesek))]
print(f'\nA többiek pozíciója a(z) {len(osveny)} hosszú ösvényen:')
[print(f'{i+1}. játékos: {poziciok[i]}. mező') for i in range(jsz) if i+1 not in nyertesek]
```

