### 1.a. példa, amikor soronként adnak meg adatokat.

szavak=list()

with open("szavak.txt") as f:

 for line in f:

 a=line.strip()

 szavak.append(a)

#print(szavak)

### 1.b. példa:

lista=[]

with open("szoveg.txt") as f:

 for sor in f:

 sz = sor.split()

 lista.append(sz)

#print(lista)

### 2. példa, amikor soronként adnak meg adatokat, de az 1. sor mást tartalmaz.

i=0

szam=0

be=list()

with open("program.txt") as f:

 for line in f:

 if i==0:

 szam=int(line.strip())

 if i>0:

 be.append(line.strip())

 i+=1

#print(szam)

#print(be)

### 2.b. példa, amelyben az 1. sor egy megoldókulcsot tartalmaz,a többi sor 2 részből áll:

# a versenyzők kódjával kezdődik - ami max. 5 karakter, ezt egy szóköz, majd az adott versenyző által adott válaszok sorozata követi.

l2=[]

meg=""

with open("valaszok.txt") as f:

 meg= f.readline().strip() # readline: egyszerre csak egy sort olvas (beleértve a sorvége karaktert is)=>

 for sor in f: # strip: eltávolítja a string elején és végén lévő betűközöket.

 l2.append(sor.split()) # split: egy stringet alakít át substringek listájává.

#print(meg) # megoldókulcsot tárolja

#print(l2) # versenyzők kódjai + válaszai

### 3. példa, amikor soronként adnak meg adatokat, de az 1-3. sor mást tartalmaz.

lift=list()

szintszam=0

csapatszam=0

igenyszam=0

with open("igeny.txt") as f:

 i=0

 for line in f:

 if i==0:

 szintszam=int(line.strip())

 if i==1:

 csapatszam=int(line.strip())

 if i==2:

 igenyszam=int(line.strip())

 if i>2:

 lift.append( [ int(e) for e in line.split() ] )

 i+=1

#print(szintszam)

#print(csapatszam)

#print(igenyszam)

#print(lift)

### Itt pedig az előző 3.példa rövidebb változata, ami ugyanúgy működik.

with open("igeny.txt") as f:

 szintszam2=int(next(f))

 csapatszam2=int(next(f))

 igenyszam2=int(next(f))

 lift2=[]

 for line in f:

 lift2.append( [ int(e) for e in line.split() ] )

#print(szintszam2)

#print(csapatszam2)

#print(igenyszam2)

#print(lift2)

### 4. példa, amikor soronként 3 adattal is találkozunk. (Mindegyik szám.)

l4=[]

with open("tavok.txt") as f:

 for sor in f:

 n,s,km = sor.split()

 l4.append((int(n),int(s),int(km)))

#print(l4)

### 5. példa, amikor soronként 5 adattal is találkozunk. (Az első kettő szám, a többi szöveges típusú.)

l5=[]

with open("szavazatok.txt") as f:

 for sor in f:

 k,sz,vn,kn,p = sor.split()

 l5.append((int(k),int(sz),vn,kn,p))

#print(l5)

### 6. példa, amelyben az F karakter azt jelzte, hogy az adott vásárlónak nincs már újabb árucikk a kosarában, fizetés következik.

l6=[]

l6seged=[]

with open("penztar.txt") as f:

 for sor in f:

 sor = sor.strip() #Eltávolítjuk vele a sorok végi ENTER-t.

 if sor != "F":

 l6seged.append(sor)

 else:

 l6.append(l6seged)

 l6seged=[]

#print(l6)

### 7. példa, Egy osztály második félévi hiányzásai állnak rendelkezésére a naplo2.txt fájlban.

#A hiányzások naponként csoportosítva szerepelnek, minden napot a # karakter kezd, majd 1-1

#szóközzel elválasztva a hónap és a nap sorszáma következik.

#Az aznapi hiányzások tanulónként külön sorokban vannak, a tanuló napi hiányzásait egy 7 karakterből álló

#karaktersorozat írja le.

l7=[] # A hiányzások hónapjainak és napjainak tárolására szolgál.

l77=[] # A tanulók neveinek és tanóráinak tárolására szolgál.

with open("naplo.txt") as f:

 for sor in f:

 adat1,adat2,adat3 = sor.split() # A txt-ben soronként 3 féle adatot látunk.

 if adat1 == "#":

 l7.append( (int(adat2), int(adat3)) ) # Hónap és nap tárolása az l2 listában.

 else:

 l77.append( (adat1+" "+adat2, adat3) ) # Nevek és tanórák tárolása az l3 listában.

print(l7)

print(l77)