$$p * S + F_s = (m_2 + m_t) * g$$

— Ezután szedessünk le — fokozatosan — a mérlegsúlyokból éppen annyit, hogy a légnyomás hatására a dugattyú kezdjen felfele mozogni. Ebben az esetben a dugattyúra ható légköri nyomást a kissé megkönnyített (m_1) teher és — a most lefelé irányuló — súrlódási erő egyensúlyozza ki:

$$p * S = (m_1 + m_t) * F_s$$

— Az egyenletrendszert megoldva a légköri nyomás kifejezése:

$$p = \left(\frac{m_1 + m_2}{2} + m_t\right) * \frac{g}{S}$$

— A dugattyú S felületét egy bizonyos V térfogathoz tartozó l dugattyúlöket megmérésével számítjuk ki: $S = V / l$.

Észrevétel:
A dugattyú kihúzásával a hengerben található levegő térfogatát (20 mm^3-ről 20 cm^3-re) több mint ezerszeresére növeltük, ezáltal a kihúzott állapotban nyomása elhanyagolhatóvá vált.

Két mérés adatai (két különböző fecskendővel):

V (cm^3)	l (cm)	S_2 (cm^2)	m2 (g)	m1 (g)	mt (g)	p (Pa)	p (torr)
10	5,05	1,98	2050	1350	240	96020	720
20	5,13	3,90	4310	2700	240	94105	706
a középérték:						95062	713

Következtetés:
Összehasonlítás végett leolvastuk egy hitelesített higanyos barométer mutatta légnyomás értékét is: $p_{bar} = 730 \text{ torr} = 730 * 133{,}3 \text{ Pa} = 97309 \text{ Pa}$

Mérésünk abszolút, valamint relatív hibáját kiszámítva:

$$\Delta p = p_{bar} - p = (730 - 713) \text{ torr} = 17 \text{ torr}; \quad \delta_p = \frac{\Delta p}{p_{bar}} = \frac{17}{730} = 0{,}023 = 2{,}3\,\%$$

Mint láthatjuk, mérési eljárásunkkal sikerült a légköri nyomás elfogadható értékét megkapnunk!

Bíró Tibor

Megjegyzések két, véletlen számokat előállító módszerhez

Az ú.n. MID_SQUARE METHOD-ot John von Neumann gondolta ki véletlen (random) számsorozatok előállítására az alábbi algoritmus szerint:

(1) Legyen $r = \overline{abcd}$ tetszőleges négyjegyű szám

(2) Emeljük négyzetre: $r^2{}_1 = \overline{klmnpqrs}$

(3) Emeljük ki ez utóbbi számból a közbülső 4 számjegyet, ezekkel alkossuk meg a sorozat következő tagját: $r_2 = \overline{mnpq}$

(4) Most a sorozat második tagját emeljük négyzetre és "belezzük ki" a négyzetét a leírt módon, és így tovább.

A módszer nem vált be a gyakorlatban: észrevették ugyanis, hogy a sorozatban csakhamar "eluralkodnak" a kicsi számok, vagyis, hogy az eljárás gyakrabban produkál 5000-nél kisebb mint annál nagyobb számokat (a számok empirikus eloszlása nem egyenletes).

Megjegyzéseim: A. Elméleti úton - mindeddig - sem cáfolni, sem igazolni nem tudom a fenti sejtést;

B. Annyi bizonyos, hogy a sorozatnak nem lehet több mint 10000 egymástól különböző tagja (hiszen csak ennyi különböző négyjegyű szám írható fel 0000-tól 9999-ig, nem több)

C. Mihelyt a sorozat valamelyik tagja egy későbbi sorszám alatt megismétlődik, attól kezdve ismétlődni fog a kettejük közti egész "szekvencia": úgy mondanám, hogy a sorozat kvázi-periodikus olyan értelemben, hogy $(\exists)$ n_o, $T \in N^*$, n_o, $T < 1000$, $(\forall) n \in N^*$, $n \geq n_o$, $r_n = r_{n+T}$.

D. Bizonyos startszámok (r_1) esetén a sorozat adott ponton "lefullad", pl. $r_1 \leq 0009$-re már a második tagtól kezdődően $r_2 = r_3 = \ldots = 0000$. Ez nyilván $T = 1$ periódusnak felel meg. Nem tudom, hogy a 0000 startszámon kívül vannak-e más olyan startszámok is, amelyekre a sorozat "önmagába fúl"?

E. Ha készíteni akarnék egy programot a Neumann-számok generálására, adott startszámra, leállíthatnám a megjelenítést mihelyt a ciklus bezárul (mihelyt tehát, "beüt" az első ismétlődés)

F. A sorozat akármelyik tagjától indítható, vagyis ha egy bizonyos startszám már fellelhető valamely előző startszám sorozatában, akkor a neki megfelelő sorozat egyszerűen átvehető az előbbiből. Ez igen jelentős időmegtakarítást jelentene.

Fentieket -és természetesen a saját észrevételeket is- felhasználva, készítsünk és futtassunk le egy programot amely mindegyik abcd startszámra kijelezné a hozzárendelt kvázi-random sorozat tagjait az első ciklus "hosszát" (a T periódust), kiírná a kicsi meg a nagy számok számarányát egy cikluson belül, azután összeszedné az egész irdatlan számtáblázatból és külön tabellálná a rendre $T = 1,2,3, \ldots$ hosszúságú ciklusokat, végül külön kiírná azokat a startszámokat, amelyekre a sorozat "lefullad" zéróra (megjelölve a lefulladási szintet/rangot is).

Feltételezésem szerint valamely transzcendens függvény értékeinek tizedesjegyei, úgy a másodiktól-harmadiktól kezdődően, már teljesen véletlenszerűen következnek egymásra. El tudnám tehát képzelni random számok generálását a következőképpen:

(1) Választok egy transzcendens függvényt (f);

(2) Az argumentumot FOR... NEXT ciklussal megfuttatom egy intervallumon, tetszőleges, kicsike növekménnyel (STEP-pel);

(3) A függvény megfelelő értékeiből leszakítom az első két tizedesjegyet, vagyis képezem a 100 * f(x) — INT (100 * f(x)) számokat.

Készítsünk programot egy-két ezer random szám előállítására ezzel az eljárással és teszteljük le, hogy egyenletes eloszlásúak-e.

Krámli József, tanár Marosvásárhely.

Megjegyzés:

Az $f:[0,1] \rightarrow R$, $f(x) = e^x$ függvény segítségével 1 000 számot generáltam, ezeket az első tizedesjegyük szerint osztályoztam és számláltam meg. A felállásuk

[0, 0,1) -on	94 szám
[0,1, 0,2) -on	103 szám
[0,2, 0,3) -on	100 szám
[0,3, 0,4) -on	97 szám
[0,4, 0,5) -on	105 szám
[0,5, 0,6) -on	95 szám
[0,6, 0,7) -on	101 szám
[0,7, 0,8) -on	107 szám
[0,8, 0,9) -on	95 szám
[0,9, 1,0) -on	103 szám volt, ami nem is olyan rossz.