Számelméleti beadandó feladatok

1. Készíts programot, amely megadja egy természetes szám összes prímosztóját!

2. Készíts programot, amely megadja a barátságos számokat 2-től N-ig (akkor barátságosak, ha az egyik osztói összege éppen a másik, a másik osztói összege pedig az egyik – az 1-et beleszámolva az összegbe, a számokat magukat pedig nem)!

3. Készíts programot, amely megadja a tökéletes számokat 2-től N-ig (akkor tökéletes egy szám, ha az osztói összege éppen önmaga – az 1-et beleszámolva az összegbe, a számot magát pedig nem)!

4. Készíts programot, amely megadja az erősen összetett számokat 2-től N-ig (akkor erősen összetett egy szám, ha az osztói száma nagyobb, mint bármely nála kisebb szám osztói száma)!

5. Készíts programot, amely megadja a 3-jegyű Armstrong-számokat (ezek olyan számok, amelyek számjegyei köbösszege éppen maga a szám)!

6. A szerencsés számokat az alábbi eljárással kapjuk. Vegyük az 1,2,3,…,N sorozatot. Ebből minden második számot törölve az 1,3,5,7,9,… sorozatot kapjuk. A megmaradt számok közül a következő, még nem használt szám a 3, így elhagyjuk a sorozat minden harmadik tagját: 1, 3,7,9,13,15,19,21…marad. Most minden hetediket kell elhagyni, s kapjuk az 1,3,7,9,13,15,21 sorozatot, és így tovább. Azokat a számokat hívjuk szerencsés számoknak, amelyek megmaradnak. Írj programot, amely kiírja az 1 és N közötti szerencsés számokat!

7. „Erősen páratlan” számoknak nevezzük azokat a páratlan számokat, amelyek vagy egyjegyűek, vagy pedig a számjegyeik összege erősen páratlan szám. Készíts programot, amely beolvas egy legfeljebb 100 jegyű természetes számot, majd kiírja, hogy ERŐSEN PÁRATLAN vagy sem! Ha nem erősen páratlan, akkor kiírja a fenti kiszámítás sorrendjében előjött első páros számot

8. Készíts programot, amely két adott természetes szám (2≤A≤2000, 2≤B≤2000) esetén megadja növekvő sorrendben, hogy az összes i természetes számra A^i B-vel osztva milyen maradékot adhat.

9. A Faktoriális(N) függvény rendkívül gyorsan növekszik. Míg az 5!=120, addig már a 10!= 73628800 ábrázolásához 4 byte-os egész számokra van szükség. A 100! azonban sem 4, sem 8, … byte-os egész számként nem ábrázolható a számítógépben. Tudjuk azonban, hogy minden természetes számnak elkészíthető a prímtényezős felbontása. Például: $5! = 2^3 * 3 * 5$, $10! = 2^8 * 3^4 * 5^2 * 7$. Készíts programot, amely kiírja az N! prímtényezős felbontását!

10. Készíts programot, amely kiírja egy természetes szám prímtényezős felbontását!

11. "Nagyon" prímeknek nevezzük az olyan prímszámokat, amelyek bármely kezdőszelete is prímszám. Például nagyon prím a 239, mert a 2, a 23 és a 239 is prím; nem nagyon prím a 241, ami ugyan prímszám, a 2 is prím, de a 24 nem az. Készíts programot, amely előállítja az összes n-jegyű nagyon prímet! A program írja ki ezen számok kezdőszeleteit is!

12. N-jegyű *társas számok*nak hívjuk a legfeljebb N-jegyű természetes számok egy a_1, a_2, …, a_n sorozatát, amelyre igaz, hogy a_1 osztói összege a_2, a_2 osztói összege a_3, …, a_n osztói összege pedig a_1. (Megjegyzés: a társas számok a barátságos számok általánosítása.) Készíts programot, amely összes N-jegyű társas szám közül azokat, amelyek legkisebbje az [X,Y] intervallumba esik!

13. Az ókori egyiptomi matematikusok a 0 és 1 közötti racionális számokat egységtörtek összegeként $1/x_1+1/x_2+\ldots 1/x_k$ alakban adták meg, ahol az x_i-k különböző pozitív egész számok.

<u>Példa</u>: $\frac{2}{5}=\frac{1}{3}+\frac{1}{15}$ $\qquad \frac{9}{11}=\frac{1}{2}+\frac{1}{4}+\frac{1}{15}+\frac{1}{660}$ $\qquad \frac{19}{30}=\frac{1}{2}+\frac{1}{8}+\frac{1}{120}$

Készíts programot, amely adott M (1≤M<N) és N (2≤N≤30) természetes számokra megadja M/N egységtörtekre bontását!

14. A Zeckendorf tétel alapján minden természetes szám egyértelműen előállítható Fibonacci számok összegeként úgy, hogy $n = F_{k_1} + F_{k_2} + \ldots + F_{k\,r}$, ahol $\forall i(1 \le i < r): k_i \ge k_{i+1} + 2$, és $k_r \ge 2$. A Fibonacci számokat az alábbi képlettel számolhatjuk:

$$F_k = \begin{cases} 0 & ha \quad k = 0 \\ 1 & ha \quad k = 1 \\ F_{k-1} + F_{k-2} & ha \quad k > 1 \end{cases}$$

Készíts programot, amely adott *n* természetes számot felbont Fibonacci számok összegére!

15. Az ún. kis Fermat-tétel azt mondja ki, hogy ha p prímszám, a pedig olyan egész szám, amely nem osztható p-vel, akkor az ap-1-1 különbség osztható p-vel Például 212 = 4096 13-mal osztva 1 maradékot ad. A kis Fermat-tétel fordítottja azonban nem érvényes, azaz ha az ap-1-1 különbség osztható p-vel, abból nem következik, hogy p prímszám. Például: $2^{340} \equiv 1 \pmod{341}$, pedig $341 = 11 \cdot 31$. Vannak olyan p összetett számok is, amelyek minden, p-nél kisebb, p-hez relatív prím a-ra kielégítik a kis Fermat-tételt. Az ilyen p-ket felfedezőjükről Carmichael-számoknak nevezzük. A legkisebb ilyen szám az 561. Készíts programot, amely beolvas két természetes számot (1≤N≤M≤100000), majd kiírja az N és M közötti Carmichael számokat!

16. „Boldog számoknak” nevezzük az olyan számokat, amelyekre igaz, hogy számjegyeik négyzetösszegét összeadva addig, amíg egyjegyű nem lesz, végül egyet (1) kapunk. Például boldog szám a 23, mert 22+32=4+9=13, 12+32=1+9=10, 12+02= 1+0=1. Készíts programot az 1 és N közötti boldog számok kiírására!

17. Készíts programot, amely kiírja az összes 2 és N közötti olyan természetes számot, amelynek semmilyen négyzetszám nem osztója!

18. „Erősen prím-összetett” számoknak hívjuk azokat a természetes számokat, amelyek prímosztói száma több, mint bármely náluk kisebb természetes szám prímosztói száma. Készíts programot, amely kiírja N-ig az erősen prím-összetett számokat!

19. Adjuk össze egy legfeljebb 100 jegyű természetes szám számjegyeit! Ha a kapott szám nem egyjegyű, akkor ennek újra adjuk össze a számjegyeit, s az eljárást folytassuk, amíg egyjegyű számot nem kapunk. Egyes kiinduló számok esetén a végeredmény 1 lesz. Nevezzük ezeket a kiinduló számokat „egyes” számoknak. Pl. 91: (91→10→1), 1998: (1998→27→9), 1999: (1999→28→10→1) Készíts programot, amely beolvas egy legfeljebb 100 jegyű természetes számot, majd kiírja, hogy a szám „egyes” szám-e!

20. Egy kiszámolós játékban N gyerek körbe áll. A kiszámolás az elsőnél kezdődik, majd minden K-adikat kell kihagyni úgy, hogy végül csak egyetlen egy gyerek maradjon. Először tehát a K. marad ki, majd a 2*K., ... Ha az utolsóhoz értünk, a kör tovább folytatódik. Készíts programot, amely beolvassa a gyerekek számát ($1<N\leq 100$) és hogy minden hányadikat kell kihagyni ($K\geq 1$), majd kiírja a kiszámolós játékban kiesőket, majd pedig a végén megmaradt gyerek sorszámát!

21. Készíts programot, amely kiírja két természetes szám közös prímosztóit!

22. A Möbius-függvény definíciója:

$$m(m)=\begin{cases}(-1)^r & ha \quad\quad m=p_1p_2\ldots p_r\\ 0 & ha \quad m \quad osztható \quad valamely \quad p^2-tel\end{cases}$$

Készíts programot a Möbius-függvény értékei kiszámítására!

23. Az N-edrendű Pierce-sorozat olyan törtek monoton növekvő sorozatából áll, ahol a nevező értéke legfeljebb N egész szám. Készíts programot az M-nél kisebb, N-edrendű Pierce-sorozat előállítására!

24. Készíts programot az N-nél kisebb olyan számok előállítására, amelyek osztói száma két prímszám szorzata!