Számelméleti algoritmusok

Bevezetés a számításelméletbe 1

4. gyakorlat

2022.

1. Határozzuk meg

   (a) 899 és 493;

   (b) 346 és 158;

   (c) 24961 és 9483

   legnagyobb közös osztóját.

2. Milyen maradékot ad

   (a) 3^{45} 79-cel osztva;

   (b) 3^{169} 91-gyel osztva;

   (c) 5^{300} 623-mal osztva;

   (d) 5^{85} 155-tel osztva?

3. Határozzuk meg a $10x \equiv 24 \pmod{m}$ lineáris kongruencia megoldásait modulo m, ahol

   (a) $m = 15$, illetve

   (b) $m = 16$.

4. Hány olyan 2012-nél kisebb pozitív egész szám van, amely 19-cel osztva 10 maradékot ad és 37-tel osztva 15 maradékot ad?

5. Legyen $n = 20181210$. Az előadáson tanult megfelelő algoritmus alkalmazásával határozzuk meg $45n + 12$ és $35n + 9$ legnagyobb közös osztóját.

6. Legyen $n = 123456$. Az előadáson tanult megfelelő algoritmus alkalmazásával határozzuk meg $12n+6$ és $9n + 4$ legnagyobb közös osztóját.

7. Egy algoritmus bemenete egy tízes számrendszerben megadott n pozitív egész szám. Az algoritmus egy ciklusból áll, aminek a magja $2n$-szer fut le, és a ciklusmag minden végrehajtásakor kiírunk egy 1-est a képernyőre.

   (a) Határozzuk meg a bemenet méretét.

   (b) Hányszor fut le a ciklusmag?

   (c) Döntsük el, hogy a ciklusmag lefutásainak száma a bemenet méretében polinomiális-e.

   (d) Határozzuk meg a ciklusmagon belül végrehajtott műveletek lépésszámát.

   (e) Határozzuk meg a teljes algoritmus lépésszámát és döntsük el, hogy ez a bemenet méretében polinomiális-e.

8. Az alábbi két C kód közül az első a bemenetként (10-es számrendszerben) kapott n pozitív egész szám négyzetét, a második pedig az n számjegyeinek az összegét számítja ki. Tegyük fel, hogy a kódok végrehajtásakor a gép az alapműveleteket az (alsó tagozatban tanult) „írásbeli” összeadás, szorzás, stb. segítségével végzi el. Döntsük el, hogy az eljárások polinomiálisak-e. (A `floor(n/10.0)` az $n/10$ alsó egészrészét adja vissza.)

(a)

```
x = n; y = 0;
while (x > 0) {
    x = x-1;
    y = y+n;
}
printf("Eredmény: %d", y);
```

(b)

```
x = 0; y = 0;
while (n > 0) {
    x = floor(n/10.0);
    y = y+n-10*x;
    n = x;
}
printf("Összeg: %d", y);
```

9. Egy képzeletbeli A algoritmus bemenete egy tízes számrendszerben megadott m pozitív egész szám. Döntsük el az alábbi állításokról, hogy igazak-e.

   (a) Ha minden m bemenet esetén A legfeljebb $5m^2$ lépés után megáll, akkor A polinomiális algoritmus.

   (b) Ha minden m bemenet esetén A legfeljebb $100 \cdot (\log_3 m)^5$ lépés után megáll, akkor A polinomiális algoritmus.

   (c) Ha minden m bemenet esetén A legalább m lépést tesz, akkor A biztosan nem polinomiális algoritmus.

   (d) Ha van olyan m, amelyre A legalább m lépést tesz, akkor A biztosan nem polinomiális algoritmus.

   (e) Ha minden páros m bemenet esetén A legalább m lépést tesz, akkor A biztosan nem polinomiális algoritmus.

10. Az alábbi C kódok közül az első $\lfloor\sqrt{n}\rfloor$-t, a második $\lfloor\log_2 n\rfloor$-t számítja ki bármely bemenetként (10-es számrendszerben) kapott $n > 0$ egész esetén. Tegyük fel, hogy a kódok végrehajtásakor a gép az (alsó tagozatban tanult) „írásbeli" összeadás, szorzás, stb. segítségével végzi el. Döntsük el, hogy az eljárások polinomiálisak-e.

(a)

```
x = 0; y = 0;
while (y <= n) {
    x = x+1;
    y = x*x;
}
printf("Eredmény: %d", x-1);
```

(b)

```
x = 0; y = 1;
while (y <= n) {
    x = x+1;
    y = 2*y;
}
printf("Eredmény: %d", x-1);
```

11. Az alábbi C kód a bemenetként (10-es számrendszerben) kapott $0 < a < n$ egészek esetén az n-nek az a-nál nemnagyobb osztói közül a legnagyobbat számítja ki. Tegyük fel, hogy a kód végrehajtásakor a gép az (alsó tagozatban tanult) „írásbeli" összeadás, szorzás, stb. segítségével végzi el. Döntsük el, hogy az eljárás polinomiális-e.

```
while (n%a != 0) {
    a = a-1;
}
printf("Eredmény: %d", a);
```