Grafika beadandó feladatok

A beadandóról általában

Részei:

1. Pascal forráskódok (főprogram, saját unitok)
2. Felhasználói dokumentáció
3. Fejlesztői dokumentáció, benne pszeudokódú algoritmus

A feladatok

1. Az y=sin(x) függvényt a (-π,π) intervallumon az alábbiak szerint rajzoljuk:

   A függvény az origóra szimmetrikus, tehát elég a [0,π) intervallumon számolni az értékeit, majd egyszerre rajzolni a kiszámolt pontot és a tükörképét. A rajzolás kezdőpontja legyen a (0,0) pont. Ha egy (x,y) pontot kirajzoltunk, akkor a következő rajzolandó pont az (x+1/S,y), (x+1/S,y+1/S), (x,y+1/S), (x+1/S,y-1/S), (x,y-1/S) pontok közül az, amelyik legközelebb van az elméleti értékhez. (S>1, egész)

   Készítsen programot, amely kirajzolja a képernyőre a koordinátatengelyeket, majd a fenti módszerrel a (-π,π) intervallumon a **színusz** függvényt, **S**-szeres nagyításban!

2. Az y=cos(x) függvényt a (-π,π) intervallumon az alábbiak szerint rajzoljuk:

   A függvény az Y-tengelyre szimmetrikus, tehát elég a [0,π) intervallumon számolni az értékeit, majd egyszerre rajzolni a kiszámolt pontot és a tükörképét. A rajzolás kezdőpontja legyen a (0,1) pont. Ha egy (x,y) pontot kirajzoltunk, akkor a következő rajzolandó pont az (x+1/C,y), (x+1/C,y+1/C), (x,y+1/C), (x+1/C,y-1/C), (x,y-1/C) pontok közül az, amelyik legközelebb van az elméleti értékhez. (C>1, egész)

   Készítsen programot, amely kirajzolja a képernyőre a koordinátatengelyeket, majd a fenti módszerrel a (-π,π) intervallumon a **koszínusz** függvényt, **C**-szeres nagyításban!

3. Az y=tg(x) függvényt a (-π/2,π/2) intervallumon az alábbiak szerint rajzoljuk:

   A függvény az origóra szimmetrikus, tehát elég a [0,π/2) intervallumon számolni az értékeit, majd egyszerre rajzolni a kiszámolt pontot és a tükörképét. A rajzolás kezdőpontja legyen a (0,0) pont. Ha egy (x,y) pontot kirajzoltunk, akkor a következő rajzolandó pont az (x+1/T,y), (x+1/T,y+1/T), (x,y+1/T) pontok közül az, amelyik legközelebb van az elméleti értékhez. (T>1, egész)

   Készítsen programot, amely kirajzolja a képernyőre a koordinátatengelyeket, majd a fenti módszerrel a (-π/2,π/2) intervallumon a **tangens** függvényt, **T**-szeres nagyításban!

4. Az y=ctg(x) függvényt a (0,π) intervallumon az alábbiak szerint rajzoljuk:

   A függvény a (π/2,0) pontra szimmetrikus, tehát elég a [π/2,π) intervallumon számolni az értékeit, majd egyszerre rajzolni a kiszámolt pontot, és a (π/2,0)-ra vonatkozó a tükörképét. A rajzolás kezdőpontja legyen a (π/2,0) pont. Ha egy (x,y) pontot kirajzoltunk, akkor a következő rajzolandó pont az (x+1/K,y), (x+1/K,y-1/K), (x,y-1/K) pontok közül az, amelyik legközelebb van az elméleti értékhez. (K>1, egész)

   Készítsen programot, amely kirajzolja a képernyőre a koordinátatengelyeket, majd a fenti módszerrel a (0,π) intervallumon a **kotangens** függvényt, **K**-szoros nagyításban!

5. A (-P,0) és (P,0) pontok által meghatározott **ellipszis**hez azok a pontok tartoznak, amelyek e két ponttól vett távolságösszege pontosan R.

   Alakítsa át a gyakorlaton megismert, pontkövető módszerű körrajzoló algoritmust ilyen ellipszis rajzolására (ismerve, hogy a következő pont merre lehet, a lehetséges következő pontok közül azt kell választani, amely távolságösszege a legkevésbé tér el R-től)!

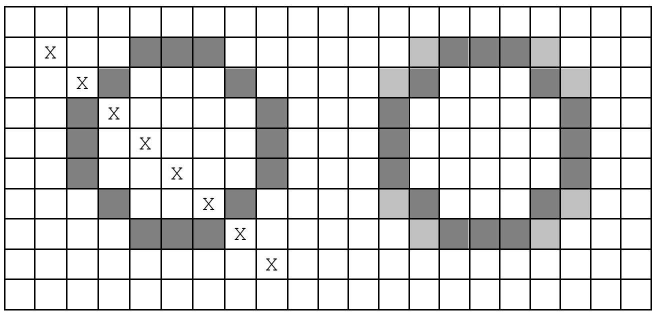
6. A (-P,0) és (P,0) pontok által meghatározott **hiperbolá**hoz azok a pontok tartoznak, amelyek távolságának különbsége e két ponttól pontosan R.

   Alakítsa át a gyakorlaton megismert, pontkövető módszerű körrajzoló algoritmust ilyen hiperbola rajzolására (ismerve, hogy a következő pont merre lehet, a lehetséges következő pontok közül azt kell választani, amely távolságkülönbsége a legkevésbé tér el R-től)!

7. Egy a gyakorlaton megismert **kör**rajzoló algoritmusban abból indulunk ki, hogy szomszédos pontokat kell rajzolni, vagy az X koordináta változott, vagy az Y, vagy pedig mindkettő.

   Alakítsa át ezt az algoritmust úgy, hogy csak egy α és β szög közötti **körív**et rajzoljon szaggatott vonallal (a két szöget az Y-tengelytől mérjük az óramutató járása szerinti irányban)! A **szaggatott vonal**at **mérete** paramétere a programnak (ennyi képpont hosszú a „vonal”, és ugyanennyi a köztük levő üres hely is). Úgy kell rajzolni, hogy az **szimmetrikus** legyen (azaz a körív két végpontjától „befelé” kell haladni a rajzolásnál; a kettő összeérésekor szabad eltérni a vonalmérettől)!

8. Egy a gyakorlaton megismert körrajzoló algoritmusban abból indulunk ki, hogy a felső pontjából indultunk, és azt néztük, hogy jobbra, jobbra lefelé, vagy pedig lefelé kell-e továbbhaladni. A **körív** egy részénél így azonban előfordulhat, hogy egy azt metsző szakas-szal nincs közös pontja. **Biztonságos**nak azt a kört nevezzük, amellyel ilyen nem fordulhat elő. Egy ilyen kört úgy kell biztonságossá tenni, hogy a körvonalon kívül levő bizonyos pontokat is befestünk (mint az ábrán látható).



9. Egy a gyakorlaton megismert körrajzoló algoritmusban abból indulunk ki, hogy a felső pontjából indultunk, és azt néztük, hogy jobbra, jobbra lefelé, vagy pedig lefelé kell-e továbbhaladni.

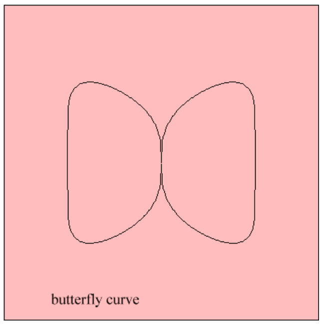
   Alakítsa át a programot úgy, hogy a rajzolással párhuzamosan **fesse** is **ki** a **kört**!

10. Egy a gyakorlaton megismert **kör**rajzoló algoritmusban abból indulunk ki, hogy szomszédos pontokat kell rajzolni, vagy az X koordináta változott, vagy az Y, vagy pedig mindkettő.

    Alakítsa át ezt az algoritmust úgy, hogy csak egy α és β szög közötti körívet rajzoljon **pontozott vonal**lal (a két szöget az X-tengelytől mérjük az óramutató járásával ellentétes irányban)! A pontozott vonal méretét képpontokban kell megadni (egy képpont hosszú a pont és méretnyi képpont hosszú a köztük levő üres hely)! Úgy kell rajzolni, hogy **szimmetrikus** legyen (azaz a körív két végpontjától „befelé” kell haladni a rajzolásnál; a kettő összeérésekor szabad eltérni a mérettől)!

11. A **pillangógörbé**t olyan $(a\cdot x, a\cdot y)$ pontok halmazaként lehet megadni, amelyeknél $y^6=x^2-x^6$ $x\in[-1,1]$.'a' szerepe: a kért rajzoló módszer számára fontos, hogy egész koordinátákkal legyenek megadva a pontok, ezért az 'a' skála-paraméterrel ($a\in\mathbb{N}$) a görbe képét egy $[-a,+a]\times[-a,+a]$ tartományban jelenítjük meg.

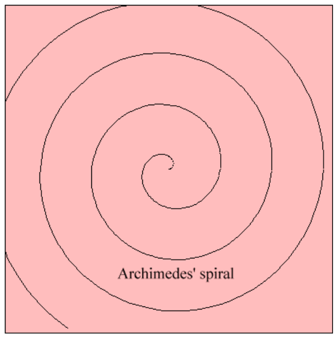
    A gyakorlaton megismertünk egy olyan körrajzoló eljárást, amely a kört szomszédos pontokkal rajzolta, minden pillanatban eldöntve, hogy a három lehetséges szomszéd közül melyiket kell következőként kirajzolni.

    Készítsen **pillangógörbé**t rajzoló programot a fenti módszerrel lényegét tekintve azonos módon!

12. Az **arkhimédeszi spirál** olyan (x,y) pontok halmazaként adható meg, ahol $x=r(\psi)*\cos(\psi)$, $y=r(\psi)*\sin(\psi)$, ahol $r(\psi)$ legyen például $\psi/10$.

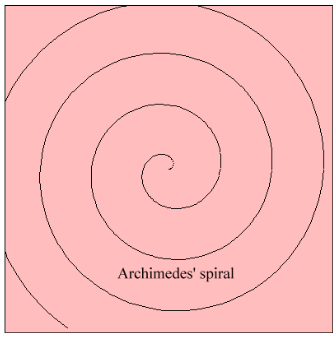
    A gyakorlaton megismertünk egy olyan körrajzoló módszert, amely a kört szomszédos pontokkal rajzolta, minden pillanatban eldöntve, hogy a három lehetséges szomszéd közül melyiket kell következőként kirajzolni.

    Készítsen **arkhimédeszi spirál**t rajzoló programot a fenti módszerrel lényegét tekintve azonos módon!

13. Az **arkhimédeszi spirál** olyan (x,y) pontok halmazaként adható meg, ahol $x=r(\varphi)*\cos(\varphi)$, $y=r(\varphi)*\sin(\varphi)$, ahol $r(\varphi)$ legyen például $\varphi/10$.

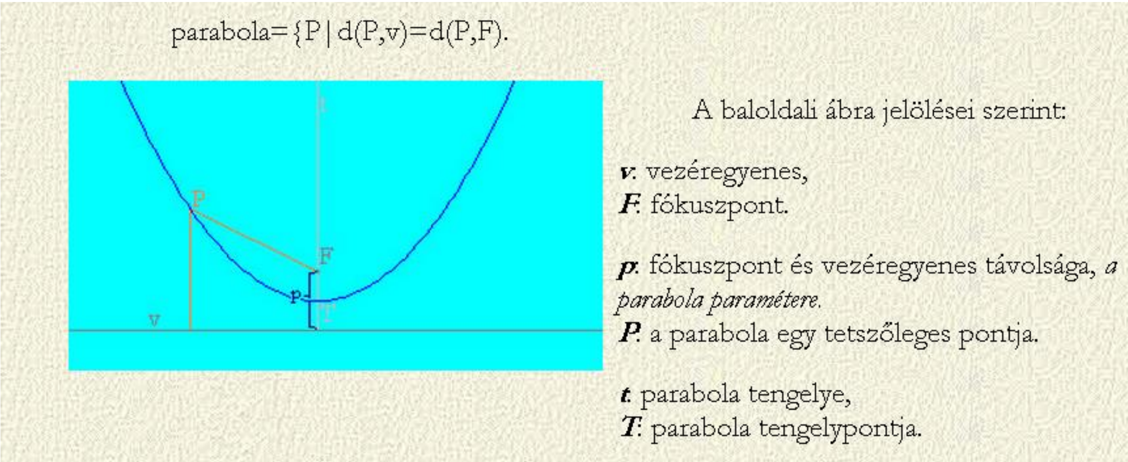
    A gyakorlaton megismertünk egy olyan körrajzoló módszert, amely lényege: egy φ szög növelésével ($\varphi\leftarrow\varphi+\Delta\varphi$) rajzolta ki a kör adott pontját. Ezt a $\Delta\varphi$ szöget kalibráltuk úgy, hogy a kirajzolt pontok érintkezzenek egymással, de minden pontot csak egyszer rajzoljunk ki. (A $\Delta\varphi$ most nem konstans, hanem egy alkalmasan választott $f(\varphi)$ függvény.)

    Készítsen **arkhimédeszi spirál**t rajzoló programot a fenti módszerrel lényegét tekintve azonos módon!

14. Egy a gyakorlaton megismert **kör**rajzoló algoritmusban abból indulunk ki, hogy szomszédos pontokat kell rajzolni, még pedig úgy, hogy vagy az X koordináta változott, vagy az Y, vagy pedig mindkettő. Tegye alkalmassá ezt a módszert az $A\cdot x^2+B\cdot x+C$ **parabola** rajzolására ($x\in[x_0-\Delta x, x_0+\Delta x]$, ahol $x_0=-B/_{2A}$, $\Delta x>0$ paraméter): tudjuk, hogy 1) a „következő" pont merre lehet, 2) a lehetséges következő pontok közül azt választjuk, amely távolsága a legkisebb a parabola adott x-hez tartozó pontjától, 3) kihasználjuk a parabola bizonyos Y-tengellyel párhuzamos egyenesre való szimmetriáját.

15. Egy a gyakorlaton megismert **kör**rajzoló algoritmusban abból indulunk ki, hogy szomszédos pontokat kell rajzolni, még pedig úgy, hogy vagy az X koordináta változott, vagy az Y, vagy pedig mindkettő. Tegye alkalmassá ezt a módszert az $A \cdot x^2 + B \cdot x + C$ **parabola** rajzolására ($x \in [x_0 - \Delta x, x_0 + \Delta x]$, ahol $x_0 = -B/2A$, $\Delta x > 0$ paraméter): tudjuk, hogy) a „következő" pont merre lehet, 2) a lehetséges következő pontok közül azt választjuk, amely távolsága a fókuszponttól megegyezik a vezéregyenestől mért távolságával, 3) kihasználjuk a parabola bizonyos Y-tengellyel párhuzamos egyenesre való szimmetriáját.



16. Az f(x)=A/x+B függvényt a [-C,C]\{0} intervallumon az alábbiak szerint rajzoljuk:

    A függvény a (0,B) pontra szimmetrikus, tehát elég a (0,C] intervallumon számolni az értékeit, majd egyszerre rajzolni a kiszámolt pontot és a tükörképét. A rajzolás kezdőpontja legyen a (C,f(C)) pont. Ha egy (x,y) pontot kirajzoltunk, akkor a következő rajzolandó pont a bal, a bal-alsó/bal-felső, ill. az alsó szomszédja közül az, amelyik legközelebb van az elméleti értékhez.

    Készítsen programot, amely kirajzolja a képernyőre a koordinátatengelyeket, majd a fenti módszerrel a (-C,C) intervallumon a fenti **hiperbolá**t!

17. Az $f(x)=A \cdot x^3 + B \cdot x$ harmadfokú függvényt a [-C,C]\{0} intervallumon az alábbiak szerint rajzoljuk:

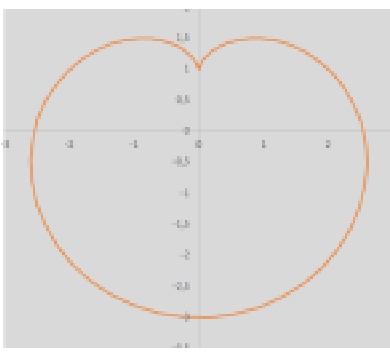
    A függvény az origóra szimmetrikus, tehát elég a [0,C] intervallumon számolni az értékeit, majd egyszerre rajzolni a kiszámolt pontot és a tükörképét. A rajzolás kezdőpontja legyen a (0,0) pont. Ha egy (x,y) pontot kirajzoltunk, akkor a következő rajzolandó pont az (x+1/H,y), (x+1/H,y+1/H), (x,y+1/H), (x+1/H,y-1/H), (x,y-1/H) pontok közül az, amelyik legközelebb van az elméleti értékhez. (H>1, egész)

    Készítsen programot, amely kirajzolja a képernyőre a koordinátatengelyeket, majd a fenti módszerrel a (-C,C) intervallumon a fenti **harmadfokú görbé**t, **H**-szeres nagyításban!

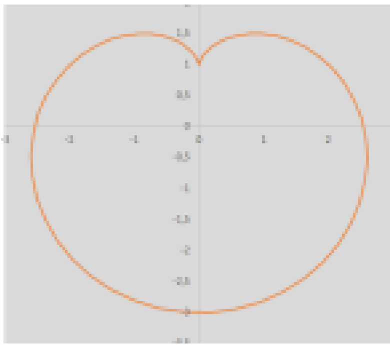
18. A **kardoid** olyan (x,y) pontok halmazaként adható meg, amely kielégíti az alábbi egyenleteket:
    $x=2*\cos(\psi)-\cos(2*\psi)$, $y=2*\sin(\psi)-\sin(2*\psi)$.

    A gyakorlaton megismertünk egy olyan körrajzoló módszert, amely a kört szomszédos pontokkal rajzolta, minden pillanatban eldöntve, hogy a három lehetséges szomszéd közül melyiket kell következőként kirajzolni.

    Készítsen **kardoid**ot rajzoló programot a fenti módszerrel lényegét tekintve azonos módon!



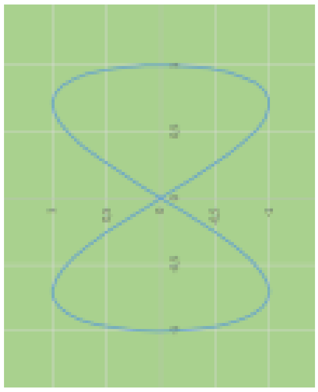
19. A **kardoid** olyan (x,y) pontok halmazaként adható meg, amely kielégíti az alábbi egyenleteket:
$x=2*\cos(\psi)-\cos(2*\psi)$, $y=2*\sin(\psi)-\sin(2*\psi)$.



A gyakorlaton megismertünk egy olyan körrajzoló módszert, amely lényege: egy φ szög növelésével ($\varphi \leftarrow \varphi+\Delta\varphi$) rajzolta ki a kör adott pontját. Ezt a $\Delta\varphi$ szöget kalibráltuk úgy, hogy a kirajzolt pontok érintkezzenek egymással, de minden pontot csak egyszer rajzoljunk ki.

Készítsen **kardoid**ot rajzoló programot a fenti módszerrel lényegét tekintve azonos módon!

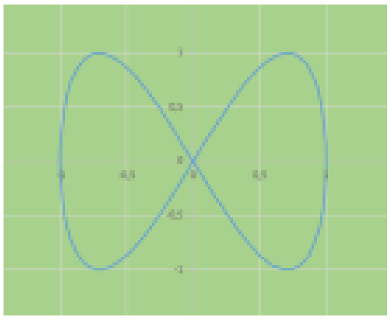
20. Egy „**oktoid**” (8-ast formáló) görbe olyan (x,y) pontok halmazaként adható meg, amely kielégíti az alábbi egyenleteket:
$x=\sin(2*\psi)$, $y=\sin(\psi)$.



A gyakorlaton megismertünk egy olyan körrajzoló módszert, amely a kört szomszédos pontokkal rajzolta, minden pillanatban eldöntve, hogy a három lehetséges szomszéd közül melyiket kell következőként kirajzolni.

Készítsen **kardoid**ot rajzoló programot a fenti módszerrel lényegét tekintve azonos módon!

21. Egy „**infinitoid**” (∞-t formáló) görbe olyan (x,y) pontok halmazaként adható meg, amely kielégíti az alábbi egyenleteket:
$x=\sin(\psi)$, $y=\sin(2*\psi)$.



A gyakorlaton megismertünk egy olyan körrajzoló módszert, amely lényege: egy φ szög növelésével ($\varphi \leftarrow \varphi+\Delta\varphi$) rajzolta ki a kör adott pontját. Ezt a $\Delta\varphi$ szöget kalibráltuk úgy, hogy a kirajzolt pontok érintkezzenek egymással, de minden pontot csak egyszer rajzoljunk ki.

Készítsen **kardoid**ot rajzoló programot a fenti módszerrel lényegét tekintve azonos módon!