**Завдання № 1**

Розробити функцію розрахунку нескладної математичної функції та оформити її у вигляді функції defun,наприклад:

1.****в інтерактивному режимі;

Варіанти:

1. y=x2 -3x+2
2. y=| 2x2 -x+1|
3. y=x2 -|x|+3
4. y=1/(x2 -3x+2)
5. y=ln(x2 -3x+2)
6. y=ln(sinx)
7. y=cos(lnx)
8. y=e^(1/sinx)
9. y=2cos(2x+1.57)
10. y=2^cosx
11. y=sin(arcsinx)
12. y=arcsin(cosx)
13. y=sinx/(x+0.5)
14. y=x+e^x
15. y=6sinx/(2+cosx)
16. y=(sinx)^4-(cosx)^3
17. y=x^3/(3-x^2)
18. y=(x-3)^2/(x+1)^2
19. y=lnx/sqrtx
20. y=(2-x)^3/(3-x)^2
21. y=2^(x^2-2x)
22. y=arcsin(sinx)
23. y=lnx/cosx
24. y=x^2/sinx
25. y=sinx/cosx
26. y=tgx/lnx
27. y=(cosx-sinx)/x
28. y=x/(lnx-cosx)
29. y=(x^2-x^3)/cosx
30. y=(x^2+x^3)/lnx

5.

**Завдання № 2.**

Виконати завдання 1 із застосуванням функцій розгалуження IF, COND,WHEN,UNLESS,CASE в залежності від значення аргументу x.

Завдання №3

Розробити функції із застосуванням управляючих конструкцій:

DO “з тілом”, DO “без тіла”, LOOP, PROG і рекурсії.

(Σ - позначка масиву суми, Π - позначка масиву помноження, ^ - позначка возведіння у степінь. Вважати першу позначку від i = 1 до i = M, другу - від j = 1 до j = N.

Пояснення: Спочатку необхідно розробити функцію розрахунку заданої формули.

Далі необхідно розробити функцію розрахунку внутрішнього циклу із застосуванням попередньої функції і без застосування. В кінцевому варіанті розробити функцію подвійного циклу із застосуванням попередніх розроблених функцій і без їх застосування.

Варіанти:

1. y = ΣΣ(i+j/i)
2. y = Σ(i!)
3. y = ΠΣ(i+ij)
4. y = ΠΣ(i/j-i)
5. y = ΣΣ(cos i + sin j)
6. y = ΣΣ(ln i / ln j)
7. y = ΠΠ(i/j – 1/j)
8. y = ΣΠ(cos i / sin j)
9. y = ΣΣ(cos i / ln j)
10. y = ΠΠ(1/i+j)
11. y = ΣΣ(log i + ln j)
12. y = ΣΠ(log i / log j)
13. y = ΣΣ(1/i^j)
14. y = ΣΣ(log i + i/j)
15. y = ΣΣ(i^j-i)
16. y = ΠΣ(i/j + sqrt j)
17. y = ΠΠ(i/ln j)
18. y = Σ(1/i!)
19. y = Π(i/j^i)
20. y = ΣΣ(sqrt i + j^2)
21. y = ΣΠ(i – j^3)
22. y = ΠΠ(j/i^2)
23. y = ΣΣ(atan i – asin(1/j))
24. y = ΠΣ(1/atan j)
25. y = ΠΠ(i/log j^2)
26. y = ΣΣ(log j-1/i)
27. y = ΣΠ(sqrt i/j+1)
28. y = ΣΣ(i^2 + sqrt j)
29. y = ΣΣ(exp i – exp j)
30. y = ΣΣ(exp i /log j)

**2 РОБОТА ІЗ СПИСКАМИ**

Із самого початку мова LISP створювалась як мова програмування, що працює із списками. Тому списки є головною структурою даних в мові LISP.

Мета виконання завдання:

практичне засвоєння техніки роботи із списками.

Завдання №4

Розробити рекурсивну функцію для роботи із списками:

1. Перемножити почленно 2 числових списка та повернути результуючий список.

2. Вибрати із списка всі числа і зібрати їх в окремий список.

3. Вибрати із списка всі елементи, які стоять на других місцях, і повернути новий

список.

4. Вибрати із списка всі треті елементи і повернути новий список.

5. Відсортувати числовий список за збільшенням.

6. Вилучити із списка заданий елемент по всьому списку..

7. Замінити в списку заданий елемент на новий по всьому списку.

8. Перемножити попарно всі елементи числового списка та повернути новий

список.

9. Скласти по 3 всі елементи числового списка та повернути новий список.

10. Визначити рівень вкладеності складного списка (максимальну кількість

входжень підсписків).

11. Із списка вилучити підсписки і повернути простий список.

12. Вилучити із списка атоми і залишити тільки підсписки.

13. Об′єднати 2 списка, взявши тільки атоми.

14. Об′єднати 2 списка, взявши тільки атоми, які не повторюються.

15. Перемножити почленно 2 числових списка і повернути результуючий список.

16. Скласти почленно 2 числових списка, взявши тільки елементи, які стоять на

других місцях.

17. Перемножити почленно 3 числових списка і повернути результуючий.

18. Зформувати із 3 списків новий, вилучивши елементи, які не співпадають. (логічне “і”).

19. Зформувати із 3 списків новий, вилучивши елементи, що співпадають

(логічне “або”)

20. Об′єднати 3 списка без повторів.

21.Додати до заданного списка елемент і поставити його за заданим по всьому

списку.

22. Зформувати із простого списка асоціативний, поставивши ключі у вигляді

порядкових чисел.

23. Зформувати із асоціативного списка новий, вибравши пари із заданим

ключом.

24. Вилучити із асоціативгого списка пари із заданим ключом.

25. Об′єднати 2 списка в асоціативний, в якому елементи 1-го списку – ключі,

2- го - об′єкти.

26. Об′єднати 3 списка в один, в якому елементи заданих списків зібрані у

трійки.

27. Відсортувати список, зібравши докупи елементи одних типів (цілі числа,

дійсні числа, символи, списки)

28. Зформувати із списка новий, зібравши докупи парні і непарні елементи.

29. Із 2-х списків вилучити елементи, які співпадають.

1. Із 2-х списків вилучити елементи, які не співпадають.

**3 ГРАФІЧНИЙ ВИВІД.**

Мета виконання завдання:

практичне засвоєння техніки програмування графічного вооду- виводу.

**Завдання №5 для виконання лабораторних робіт**

Розробити програми побудови графіків мовами COMMONLISP i AutoLISP

Приклади програм функції y= sin x:

COMMONLISP:

;=========================================

; **Функція** **розрахунку y = (sin x)**

;Вхідні: xnach -початкове значення x(рад) xkon-кінцеве значення x(рад)

; deltax -крок по x mahtab - масштабний множних для y

; Вихідні: список sx - список абсцис список sy - список ординат

;

(defun sinn (xnach xkon deltax mahtab)

(setq sx (cons xnach nil) sy (cons (\* mahtab(sin xnach)) nil))

(do ((x1 xnach)(x2 xnach))

((> x2 xkon) nil)

(setq x2 (+ x2 deltax) y2 (\* mahtab (sin x2)))

(setq sx (cons x2 sx) sy (cons y2 sy))

);зачинено do

(setq sx (reverse sx) sy (reverse sy))

); зачинено функцію sinn

;==========================================

; **Функція** **прорисовки графіка**

; вхідні: список sx - список абсцис список sy - список ординат

; вихідні: графік на екрані

;

=========================================================

(defun graf (sx sy)

((null sx) nil)

(plot-line (truncate(car sx)) (+ (truncate(car sy))170) (truncate(cadr sx)) (+(truncate(cadr sy))170) 15)

(graf (cdr sx) (cdr sy))

)

ГОЛОВНА ПРОГРАМА

(load common87)

(sinn 0 (\* 2 (pi)) (/ (pi) 10) 10)

(video-mode 16)

(plot-line 0 170 640 170 15); прорисовка вісі x

(plot-line 0 0 0 170); прорисовка вісі y

(graf sx sy)

AutoLISP:

(defun sinn (xnach xkon deltax)

(setq N (fix (/ (- xkon xnach )deltax)

(setq t1 (list xnach (sin xnach)))

(command "pline"t1)

(repeat N (setq x2 (+ x1 deltax) y2 (sin x2) t2 (list x2 y2))

(command t2)

(setq x1 x2)

)

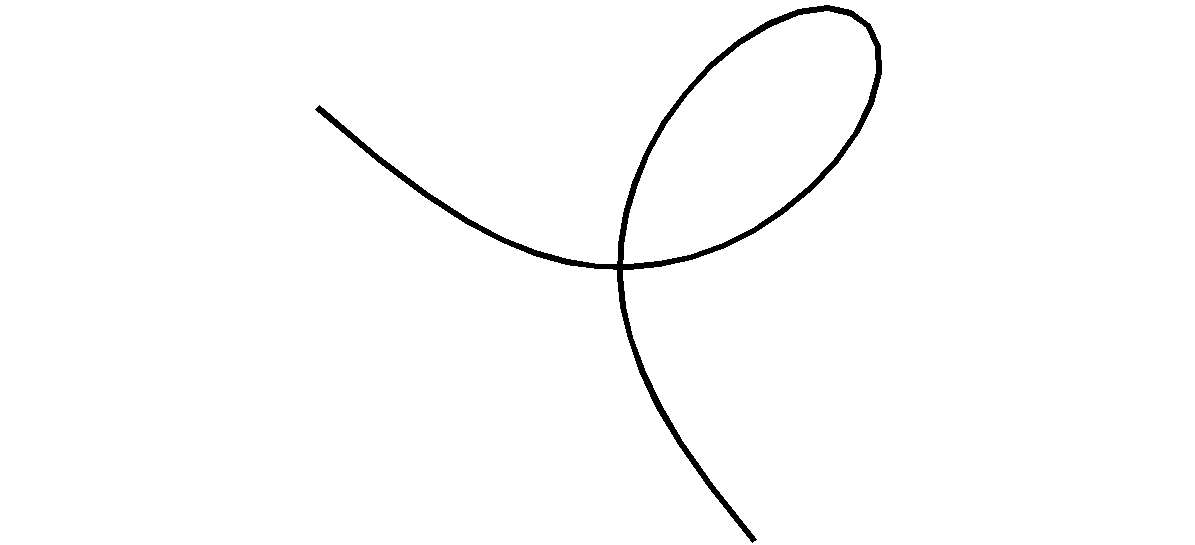
(command " ")

)

**Варіанти:**

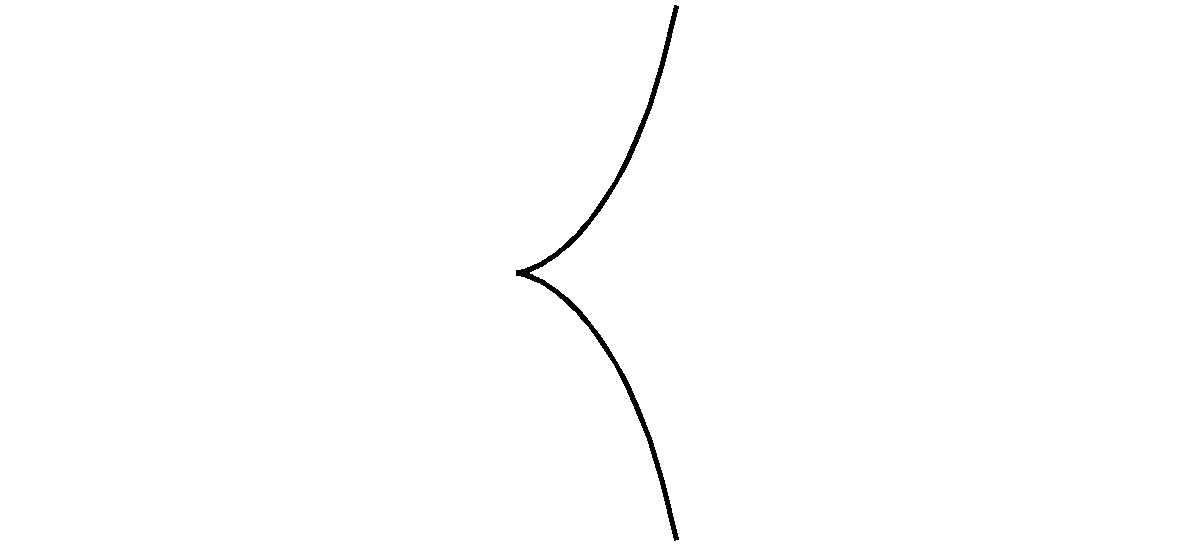
1. Декартов лист.





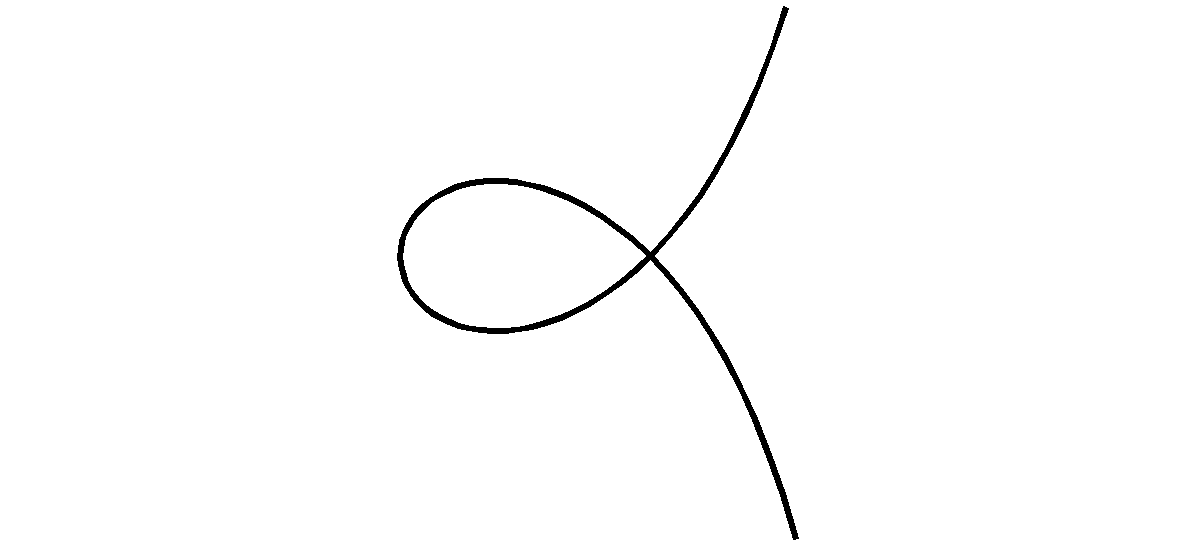
1. Циссоїда Діоклеса.





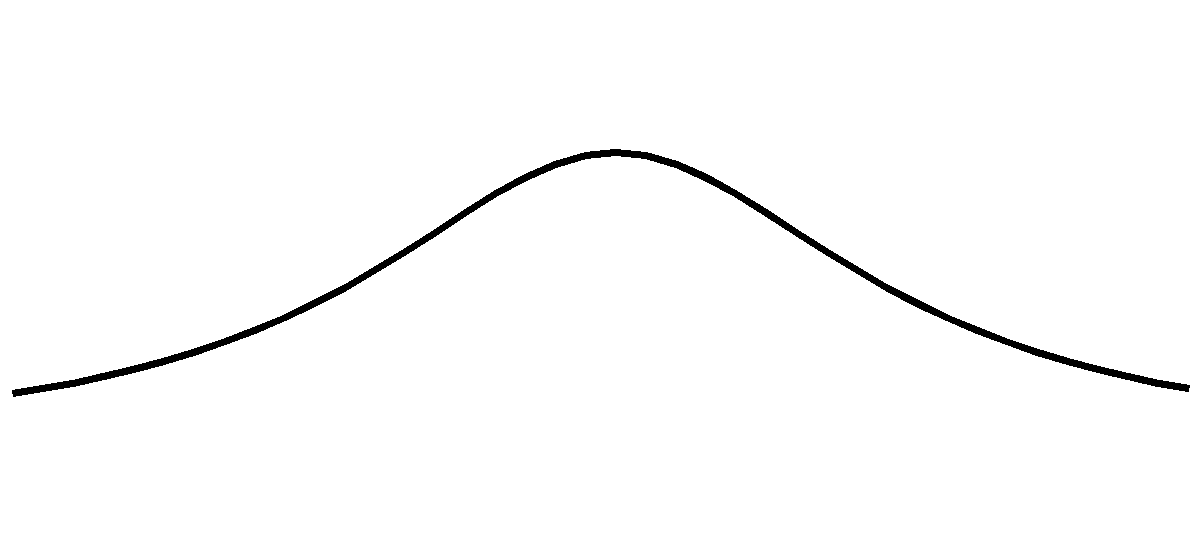
1. Строфоїда.





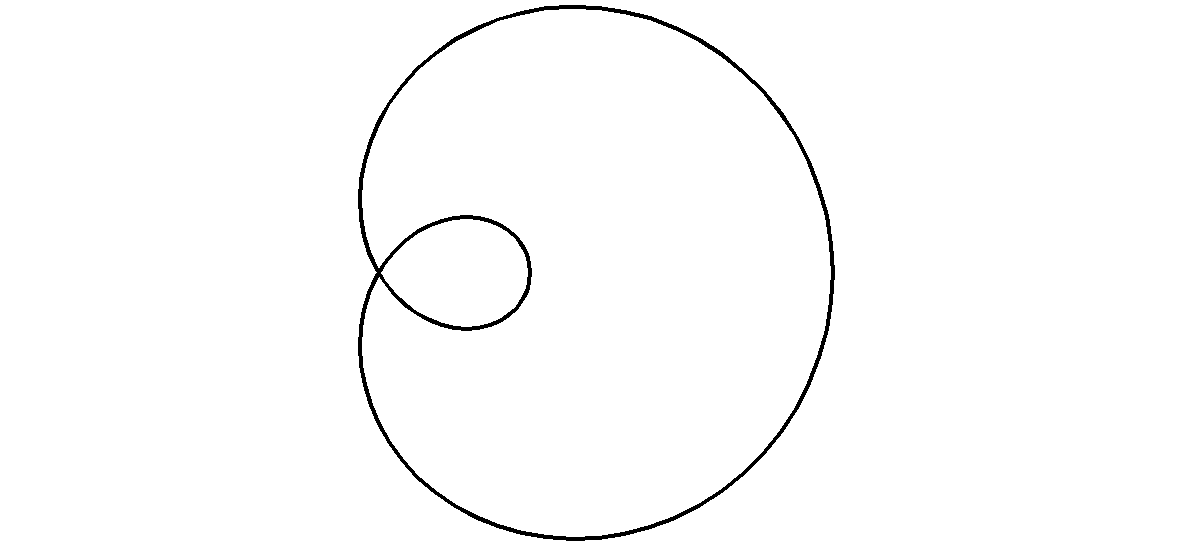
1. Верзиєра.





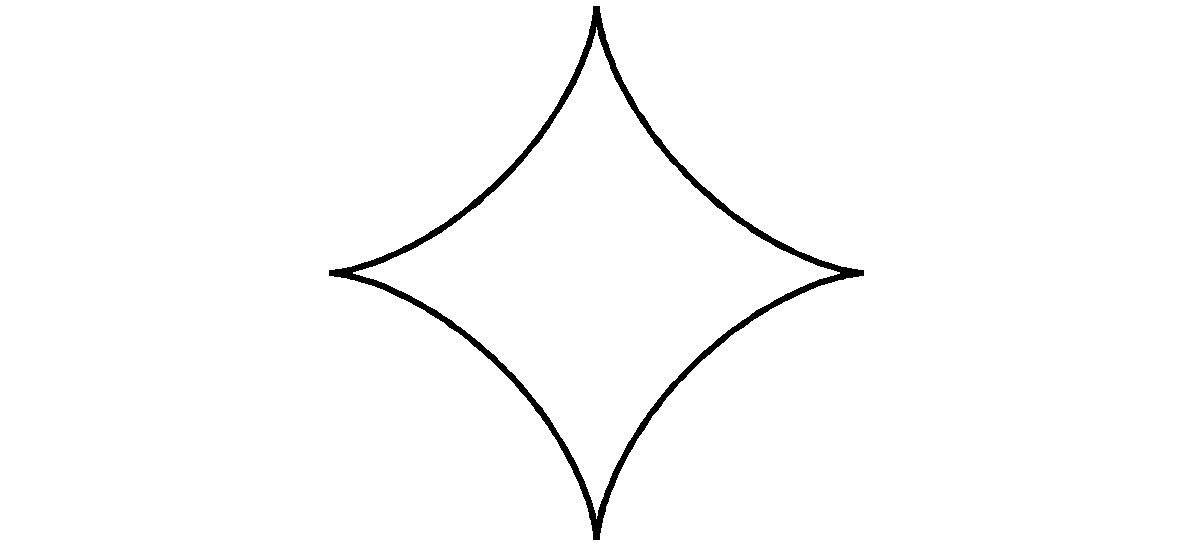
1. Улітка Паскаля.





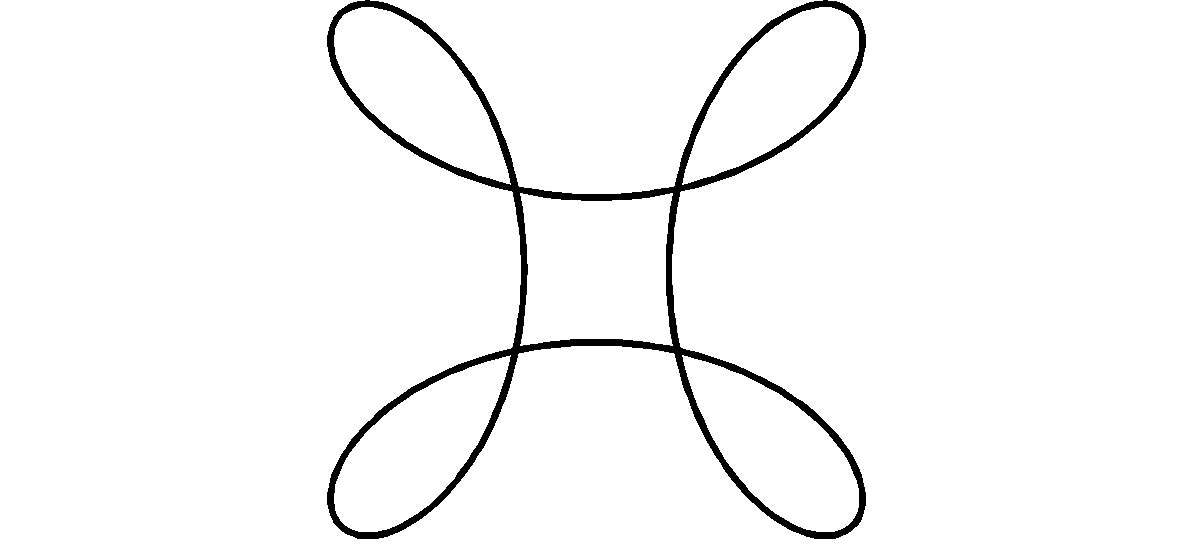
1. Епіциклоїда.





1. Епітрохоїда.

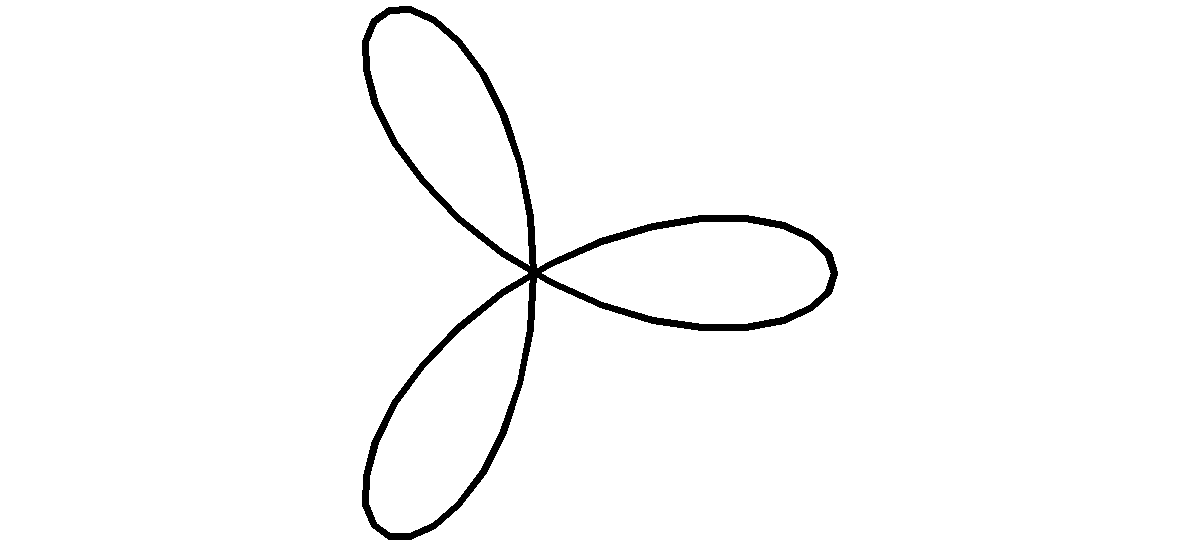




1. Подера Штейнера.

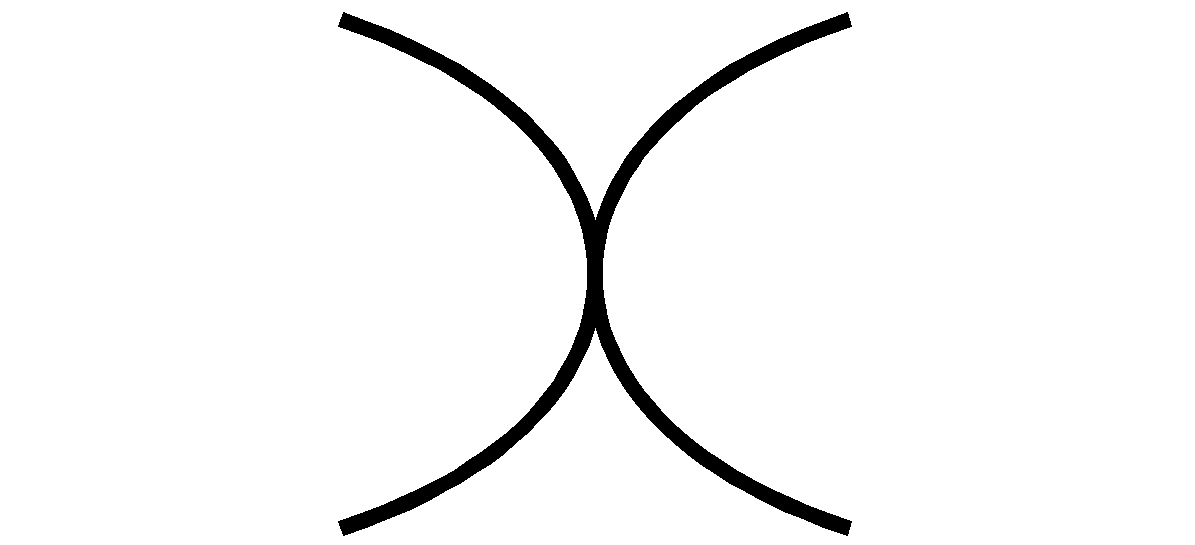


0,



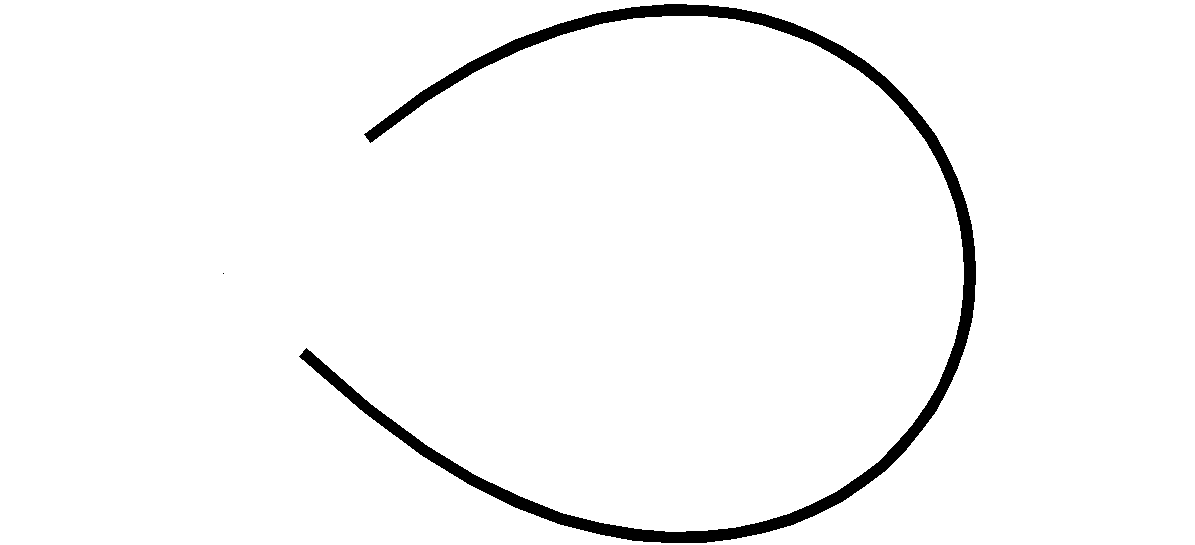
1. Крива Карра.





1. Овал Кассіні.

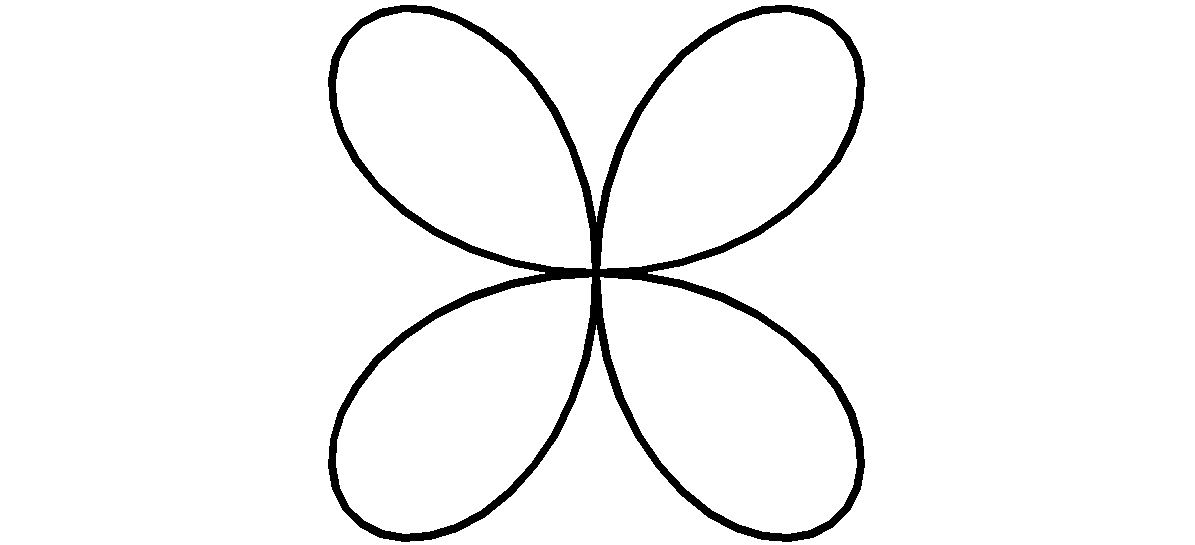




1. Роза-4.

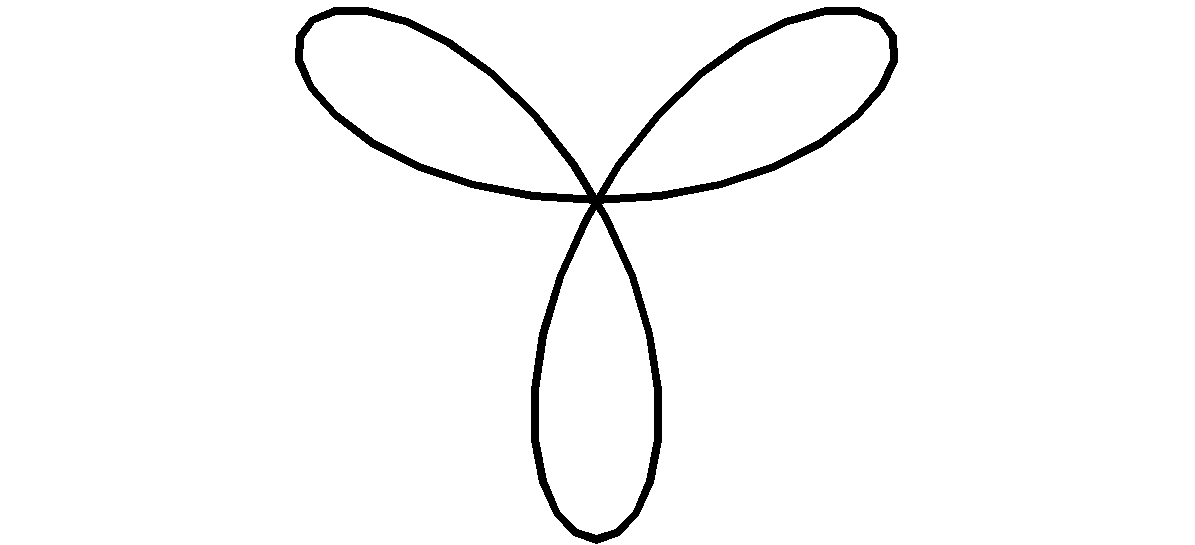






1. Роза-3.

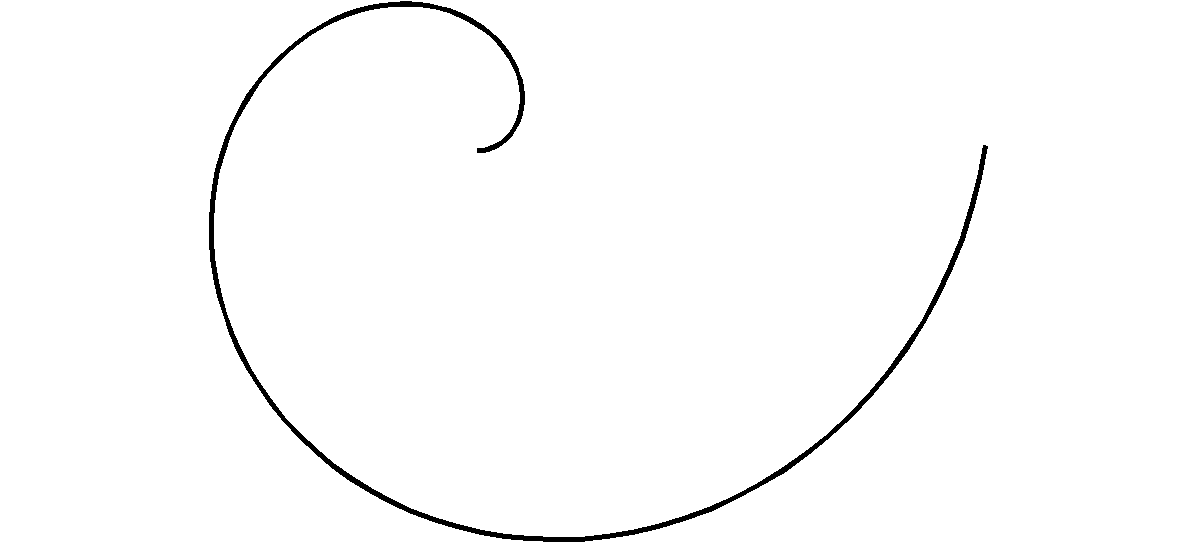




.

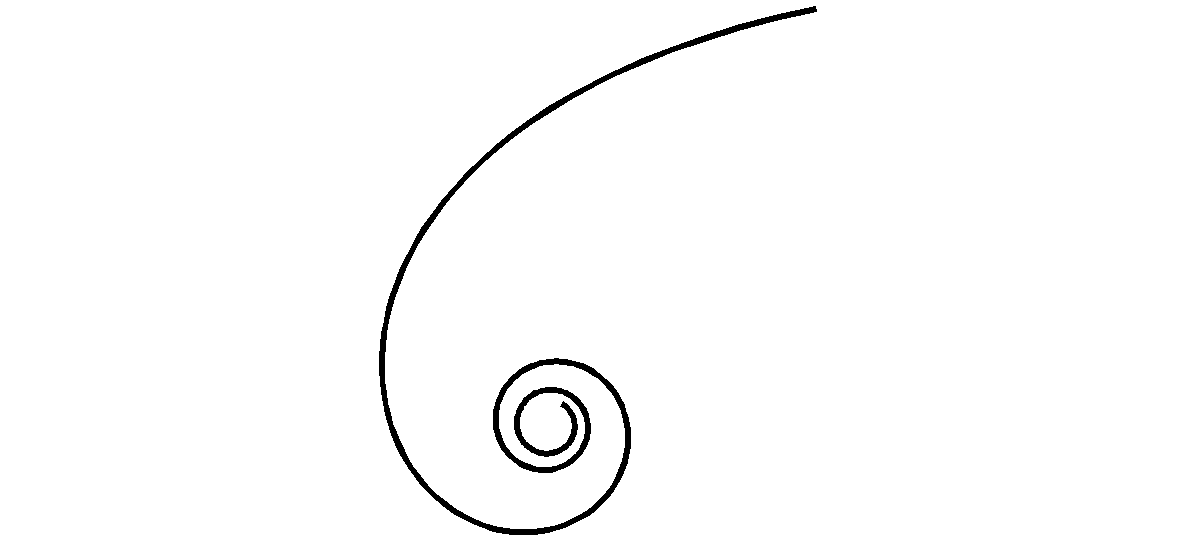
13.Спіраль Архімеда.





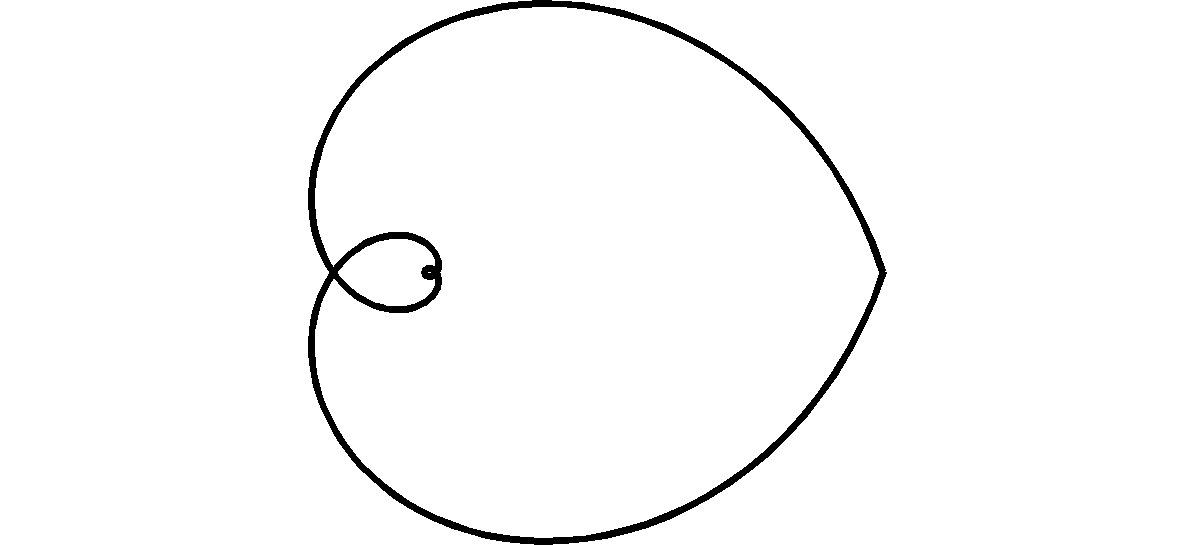
14.Гіперболічна спіраль.





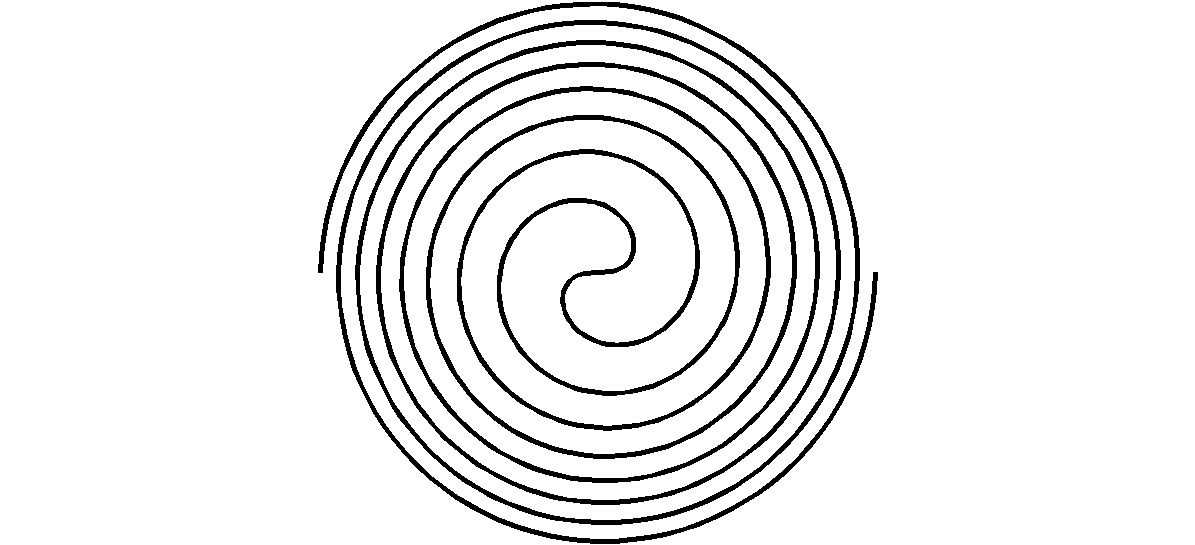
15.Спіраль Галілея.





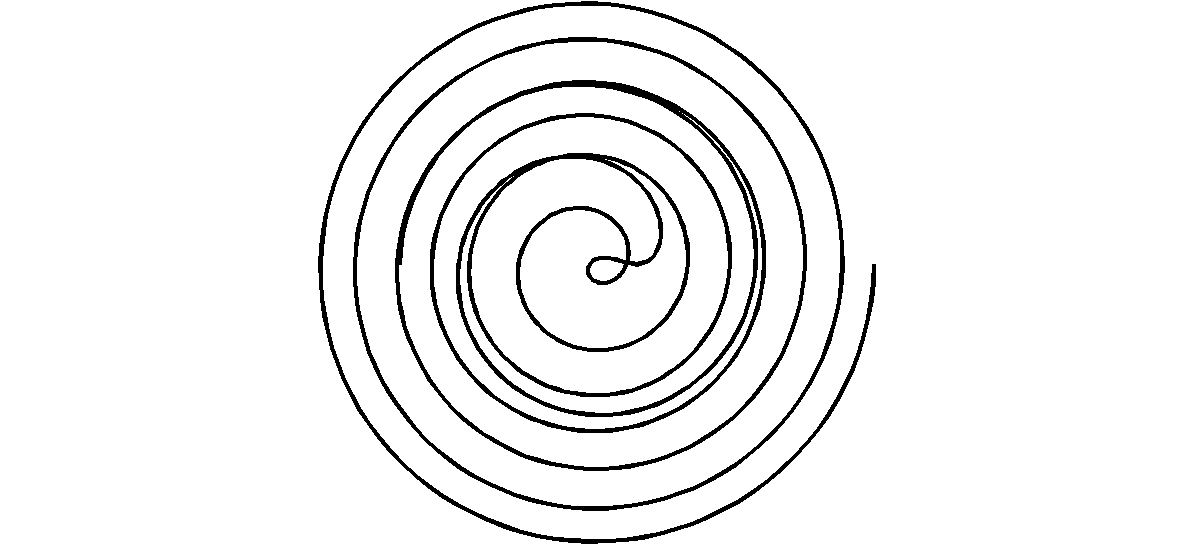
16.Спіраль Ферма.





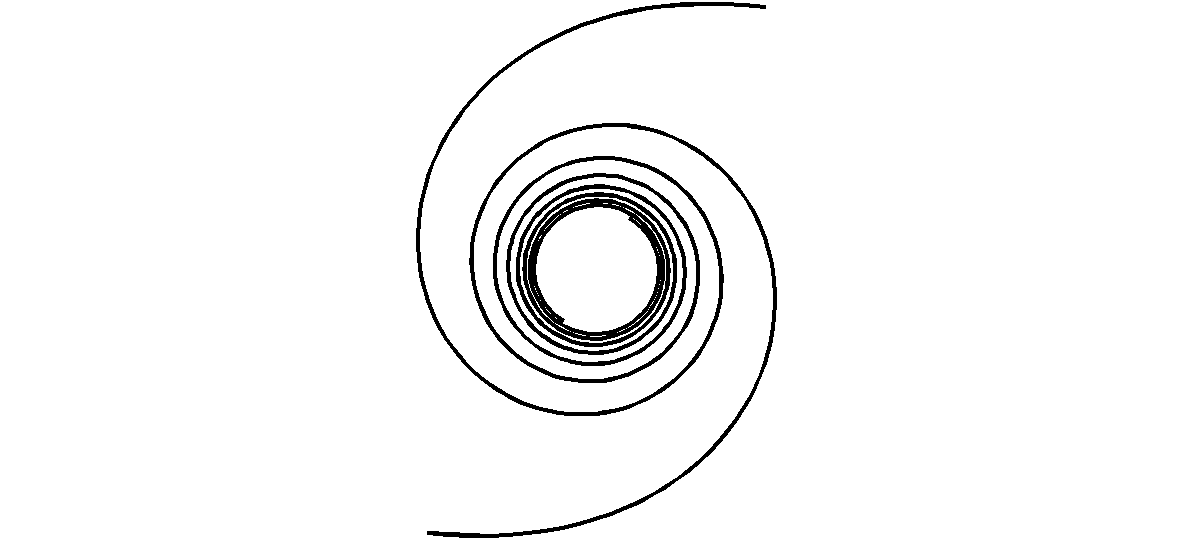
17.Параболічна спіраль.



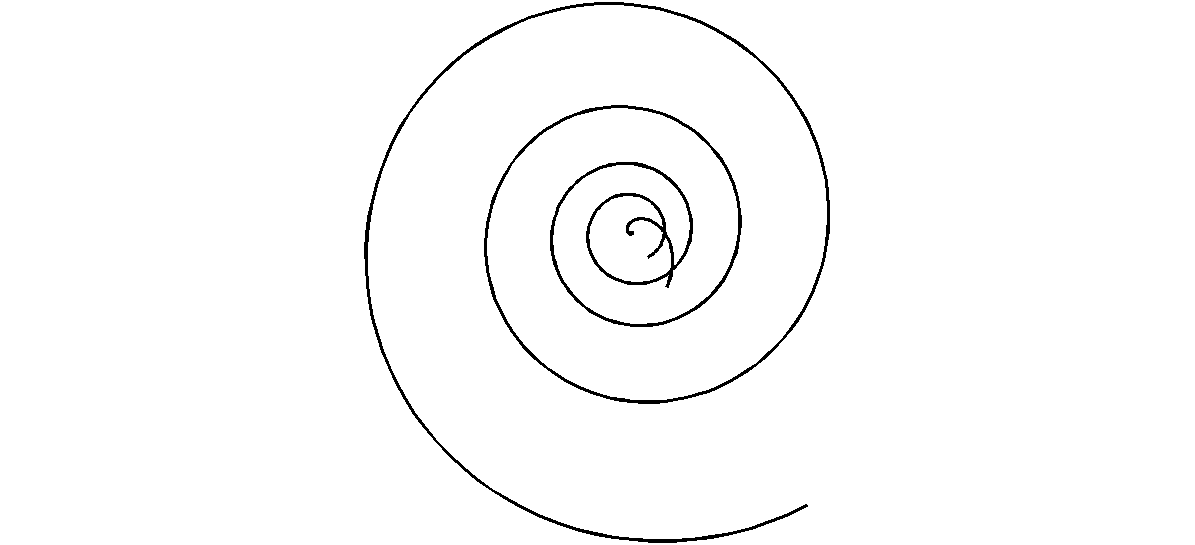


18.Спіральна булава.



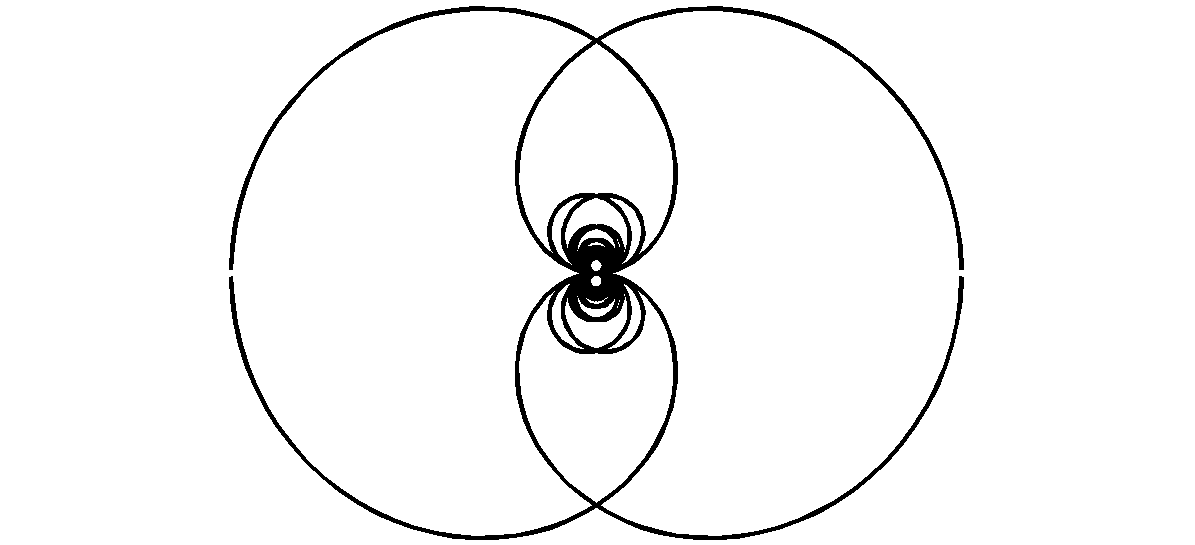


19.Логарифмічна спіраль.



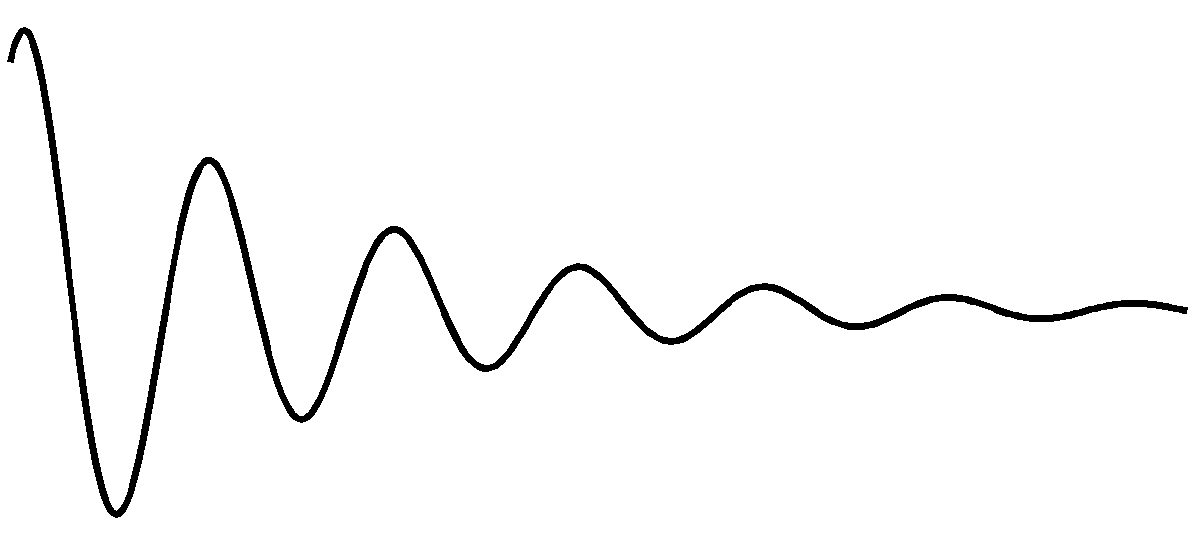
20.Кохлеоїда.





21.Затухаюча синусоїда.





22.Візерунок (“узор”).



