**Przygotowanie obrazka:**

Rozpoczynamy od zbinaryzowania obrazu. Binaryzacja musi być dokładna, więc zalecamy ręczne dobranie progu binaryzacji lub użycie obrazu z jednolitym białym tłem. Binaryzacja nie obsługuje wielu obiektów na obrazie, jednak można wyciągnąć pojedynczy obiekt, przy użyciu funkcji nadającej etykiety.  
Następnie wypełnimy wszystkie dziury w środku naszego obiektu i wykonujemy otwarcie i zamknięcie, aby pozbyć się pojedynczych pikseli.  
Kolejnym krokiem jest użycie funkcji edge(), która zwróci nam tylko krawędzie obrazu oraz obliczenie centrum masy naszego obiektu (centrum masy w tym przypadku będzie średnią arytmetyczną współrzędnych x i y).  
Następnie dla każdego piksela na krawędzi, zamieniamy jego współrzędne na współrzędne biegunowe ze środkiem masy jako punkt (0, 0). Dzięki temu możemy w łatwy sposób ustawić nasze punkty w kolejności rysowania - wystarczy, że posortujemy po thecie. Dla niektórych obrazów będzie to jednak powodowało problemy. Gdy dwa kąty są zbliżone,  
a między punktami znajduje się przerwa (np. dla wiewiórki - pomiędzy plecami a ogonem znajduje się przerwa) algorytm nie zadziała poprawnie, co zauważymy przy rysowaniu.  
Na końcu wracamy współrzędnymi do współrzędnych kartezjańskich i zapisujemy je do dwóch wektorów X i Y.

**Transformata Fouriera:**

Zaczynamy od obliczenia dyskretnej transformaty Fouriera, aby to zrobić, musimy zapisać nasze współrzędne jako liczby zespolone. Z transformaty otrzymujemy: częstotliwość, promień oraz fazę.  
Następnie wszystkie te elementy sortujemy po promieniu, ponieważ podczas rysowania chcemy rozpoczynać od największego koła i przechodzić po coraz mniejszych kołach.

**Rysowanie:**

Ustalamy krok czasu, o jaki będziemy się przesuwać z każdą iteracją. Tworzymy tablicę, w której będziemy przechowywać wszystkie elementy naszej krzywej, aby nie utracić zawartości przy przerysowywaniu, gdyż musimy czyścić „płótno“.  
Zaczynamy od obliczenia środków każdego koła i zapisujemy je w tablicy. Zapisujemy również linie (od środka poprzedniego koła do środka aktualnego koła), które będą promieniem koła, co poprawi czytelność animacji. Następnie za pomocą funkcji viscircles() rysujemy wszystkie koła i używamy funkcji „hold on“, aby przy rysowaniu promieni i krzywej, płótno nie zostało wyczyszczone. Kolejnym krokiem jest narysowanie promieni dla każdego koła, a następnie samej krzywej. Na końcu zaznaczamy punkt, w którym aktualnie rysujemy, za pomocą czerwonej kropki.