Se considera directia de mers a ramei doar directia capului ramei (sageata neagra), exprimata in radiani.

Dintr-un punct pivot se considera perpendiculara (linia portocalie) care defineste centrele cercurilor de Bounce (albastre). Altfel, cercurile de Bounce sunt definite ca fiind tangente la directie, dar cu offset astfel incat cercul va contine capul ramei in interior.

In momentul in care ambele cercuri intersecteaza marginea ecranului (se poate calcula ca distanta de la centrul cercului pana la margine), rama va urma traseul descris de cercul cel mai indepartat, sau oricare cerc daca distantele sunt egale de la centrele cercurilor la margine.

Folosind constantele SCREEN\_WIDTH si SCREEN\_HEIGHT ca dimensiuni ale ecranului, si variabilele din clasa Worm ptHeadPos, xDirection, xWormStretch, xBounceRadius scrieti o functie in clasa Worm:

double Worm::BounceScreen()

Care intoarce 0 daca unul din cercuri nu intersecteaza marginea (marginile).

Daca ambele cercuri intersecteaza marginea, ptr cel mai indepartat cerc, sa se returneze valoare negativa ptr cercul din stanga (fata de directia de mers) si valoare pozitiva ptr cercul din dreapta.

Valoarea (in modul) reprezinta adaosul la directia ramei astfel incat urmatoarea pozitie a ramei va avea cercul respectiv in acelasi loc (adica rama se roteste in jurul centrului cercului cel mai indepartat de margine). Distanta parcursa de rama la un pas este definita ca xWormStretch in clasa Worm.

Centrele cercurilor de Bounce vor fi stocate in variabilele ptLeftBounceCircle si ptRightBounceCircle dupa calculul din functia BounceScreen.