דוח הנחיות להוספת SoftShadows

מטרה: לאפשר חישוב צל רך הנובע ממקור אור בעל שטח, באמצעות שליחת קרני דגימה רבות מכל נקודת השקה לעבר מקור האור.

שלב 1- הוספת מחלקות SphereLight,DiskLight היורשות מ – PointLight,SpotLight  
בכדי להוסיף מקורות אור בעלי שטח שיוכלו לאפשר צללים רכים בהם מוסיפים משתנה רדיוס.

נוסיף גם מתודת חישוב של נקודה אקראית מהמשטח (בשביל ריבוי קרני דגימה) במחלקת LightSource וכן בכל המחלקות המיישמות אותה בשם GetRandomPointOnSurface() שתחזיר נקודת מיקום אקראית על מקור האור, נכון לעכשיו המתודה תחזיר NULL.

שלב 2 – נוסיף ב – renderer מחלקה חדשה בשם SoftShadowsUtils המחלקה תעזור ליצירת קרני דגימה באופן אקראי ממקור האור לנקודה. בתוך המחלקה ניצור מתודת generateBeamToArea שמקבלת מקור אור, נקודה וכמות קרניים. המתודה תחזיר רשימת כיוונים ממקור האור לנקודה.

שלב 3 – ב- SimpleRayTracer נוסיף שדות פרטיים חדשים של softShadowsEnabled לשימוש בSoft shadows שתאותחל כברירת מחדל ב- false. ונוסיף מתודה setSoftShadows בכדי להשתמש בSoft Shadows. בנוסף נוסיף שדה פרטי סטטי של כמות קרני דגימה (NUM\_SAMPLES).

בנוסף למתודה transparency נוסיף מתודה חדשה שנקראת softTransparency המתודה תבדוק האם יש שימוש בריבוי קרני דגימה, במידה ואין היא תשתמש ב- transparency, אחרת היא תחשב בנפרד לכל קרן דגימה את הצבע של הפיקסל ואז תחלק בכמות הקרניים על מנת למצוא את הצבע הממוצע (זה מה שיגרום לאפקט כיון שליד הגבול של ההצללה ישנם קרניים שגורמות להצללה וישנם קרניים שלא).

ב- calcColorLocalEffects נשנה את ktr כך:  
Double3 ktr = softShadowsEnabled ? softTransparency(intersection, *NUM\_SAMPLES*) : transparency(intersection);  
על מנת שנשתמש בחישובים לעיל.

שלב 4 – כעת נוכל לכתוב את המתודה GetRandomPointOnSurface למחלקות DiskLight, SphereLight שיחשבו קרן מנקודה אקראית על מקור האור.

נוסיף מחלקת בדיקה SoftShadowsUtilTest ונבדוק שאנחנו מקבלים את התוצאות הרצויות.