

דו"ח פרויקט

רקע כללי לפרויקט:

מטרת הפרויקט הינה יצירת תוכנית לדימוי סצנה גרפית וירטואלית תלת-ממדית. כלל הפרויקט נכתב בשפת JAVA תוך שמירה עבודה לפי עקרונות עיצוב והנדסת תכנה הנלמדו בקורס התאורטי. הפרויקט נכתב בליווי בדיקות (TESTING) לפונקציות העיקריות כולל בדיקות מחלקות שקילות (Equivalence Partitions Tests) ובדיקות מקרי קצה (Boundary Values Tests).

הפרויקט מורכב ממספר חבילות (PACKAGES) באמצעותם נממש את המודל ליצירת תמונה תלת ממדית להלן פירוט החבילות והמחלקות:

-PRIMITIVE

מכילה את המרכיבים הבסיסיים להגדרת מרחב R3.

COORDINATE - הגדרת קואורדינטה, נקודה באחד משלושת המישורים שבמרחב התלת ממדי.

POINT3D - נקודה במרחב מורכבת משלוש קואורדינטות

VECTOR - ייצוג ווקטור במרחב, מיוצג על ידי נקודה POINT3D שמגדירה את ראש הווקטור ועל ידי כך את הכיוון שלו.

RAY - קרן, קרן הינה ווקטור שמוגדר בנוסף על ידי מיקום התחלה כלומר הכיוון של הקרן מיוצג על ידי וקטור ובנוסף קיימת נקודה POINT3D שמגדירה את המיקום ההתחלתי שלו.

COLOR - הגדרת צבע לכל נקודה במרחב כלומר אם ישנו גוף במרחב אזי הגוף מסוגל לפלוט צבע מסוים ולכן נגדיר צבע לגופים, תאורות, ובאופן כללי נוכל להגדיר צבע רקע לסצנה עצמה או להגדיר העדר צבע כלומר צבע שחור.

MATERIAL - חומר, נגדיר סוג חומר לגופים כלומר מה רמת התגובה של הגוף בפגיעת קרן אור מסוימת הגדרה זו תתבצע על ידי מקדמי הנחתה לאפקטים של החזרת אור, מקדמי שקיפות והשתקפות ומידת הנצנוץ של החומר.

-GEOMETRIES

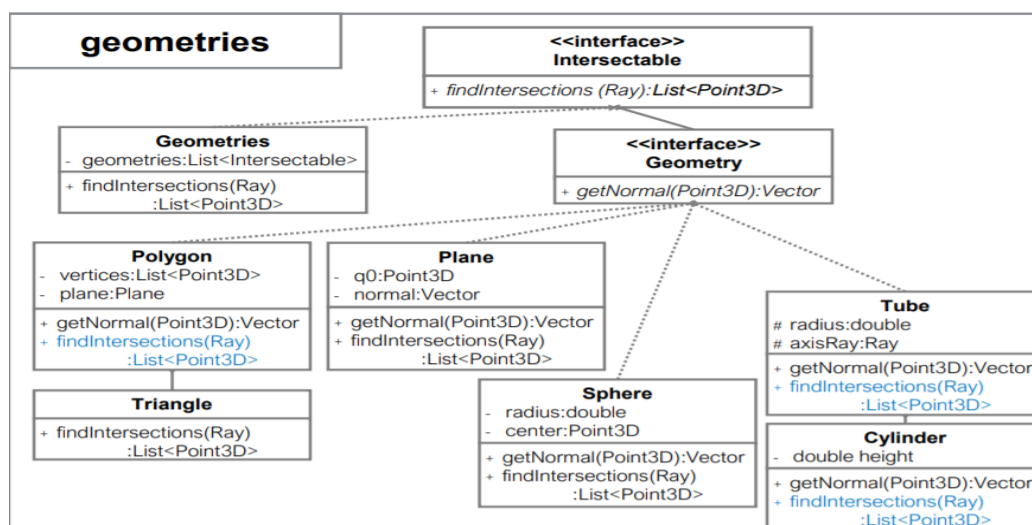
מכילה את הצורות הבסיסיות שנוכל לשתול בסצנה.

כלל הצורות יורשות מהמחלקה האבסטרקטית GEOMETRY אשר מממשת את הממשק-Intersectable .

הפונקציה העיקרית שהמחלקות דורסות היא findGeoIntersections שמחזירה את נקודות החיתוך של הקרניים שנבנה מהמצלמה ושיעברו את המשטח (view plane) עם הגופים בסצנה.

בנוסף כל מחלקה שמייצגת גוף מסוים יודעת להחזיר את הווקטור הנורמלי (אנכי) לגוף זאת על מנת לחשב חיתוכים ואינטראקציות עם תאורה.

היררכית חבילת GEOMETRIES



-ELEMENTS

כפי שהוזכר לעיל אנו בונים סצנה גרפית על ידי משטח צפייה VIEW PLANE שמאחוריו ממוקמת מצלמה או לחילופין העין של המתבונן בסצנה.

נגדיר בחבילת זו את המחלקה CAMERA ובנוסף את כלל סוגי התאורה אשר נתייחס אליהם בפרויקט.

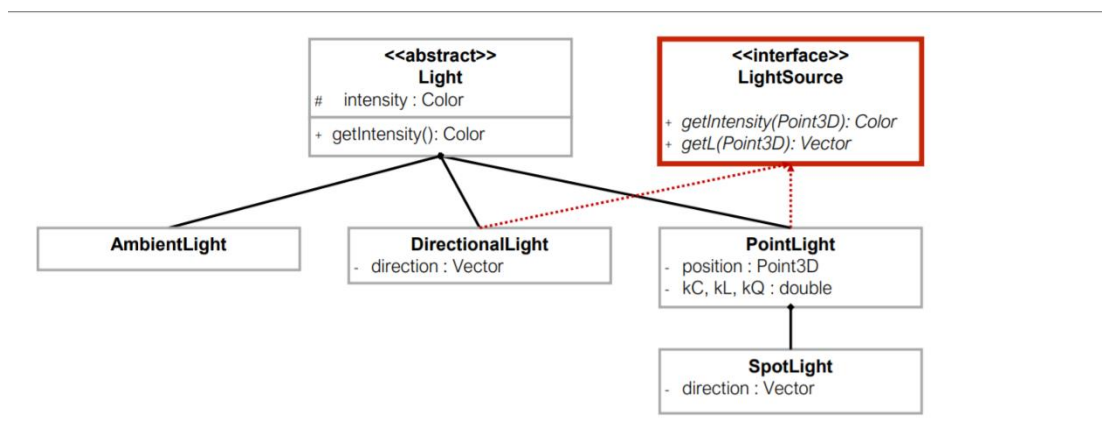
CAMERA - מורכבת מנקודה שמגדירה את המיקום של מרכז העדשה ובנוסף שלושה וקטורים – RIGHT, TO, UP שמגדירים את כיוון העדשה ואת התזוזה של על הצירים.

הממשק LIGHTSOURCE –

כלל סוגי התאורה חוץ מתאורה סביבתית AMBIENT ממשקים ממשק זה הכולל את הפונקציות

getIntensity שמחשבת את עומת האור שמגיעה ממקור האור לנקודה ולנקודה getLI שמחזירה את הוקטור ממקור האור לנקודה המוארת.

היררכית חבילת ELEMENTS



-RENDERER

חבילה זו מכילה את המחלקה העיקרית RENDER שמחזיקה את כל הנתונים הרלוונטיים לסצנה.

בנוסף קיימות המחלקות IMAGE WRITER שמייצרת קובץ JPG על ידי מטריצת צבעים שנבנית בתהליך הרנדור , RAY TRACER BASIC כלל החישובים להשגת צבע בכל נקודה בסצנה מתבצע במחלקה זו על ידי התחשבות באפקטים גלובליים , שקיפות והשתקפות (מראה) של חומרים ואפקטים לוקלים של תגובת החומר בפגיעת קרן אור והטלת הצל.

הפונקציה העיקרית CALC COLOR הינה פונקציה שתפעל באופן רקורסיבי ותבנה קרני שקיפות והשתקפות ובכך תדמה לנו אפקטים של מראה או חומר שקוף שהצבע עובר דרכו.

-UNITESTS חבילת הבדיקות ויצירת תמונות מבחן במהלך הפרויקט.**לאחר שהסברנו בקצרה על הפרויקט נפרט על שיפורי התמונה שביצענו למיני פרויקט 1:**

מטרת מיני פרויקט 1: שיפור התמונה על ידי מתן מראה מציאותי יותר.

SOFT SHADOW

הבעיה: הצל בתמונות נראה לא אמיתי מכיוון שהוא נראה כצל קשה ובעל מרקם אחיד בכל נקודה שמוצלת על ידי מרכז מקור התאורה, למרות שיתכן שבמציאות הוא עדיין מואר חלקית.

פתרון: במקום להתייחס למקור אור כנקודה אחת במרחב נתייחס למקור האור כגאומטריה בעלת שטח במרחב. מכיוון שהשתמשנו במודל שלנו ב point light ו spot light בחרנו להשתמש בעיגול דו ממדי ובמעגל תלת ממדי בהתאמה. כעת נשלח מספר קרני צל אל מספר אזורים שונים במקור האור ואחוז התאורה בכל נקודה יהיה היחס בין מספר הפגיעות במקור האור לכמות קרני הצל שיוצרו.

מימוש:

למקור האור הוספנו רדיוס שייצג את מקור האור כעיגול, הרדיוס מאוחסן בכל מחלקה של מקור תאורה, שהרי זה מאפיין של מקור האור (למעט תאורת DIRECTIONAL LIGHT שאין לה מיקום וצורה כלל).

לאחר מכן יצרנו נקודות המפוזרות במידה אחידה בשטחו של מקור התאורה. ב spot light המקור עצמו הוא מעגל. ב point light המקור עצמו הוא כדור מבחינה מציאותית, אבל כדי לחשב נקודות מספיק ליצור חתך של מעגל בתוך הכדור. המעגל נבחר כך שהאנך אליו הוא וקטור מקור האור (כלומר הווקטור ממרכז מקור האור אל הנקודה המוארת). כל קרן הצללה שניצור מהנקודה למקור האור חייבת לעבור במעגל ולכן מספיק ליצור מעגל ואין צורך בחיפוש נקודות בתלת ממד. [בנוסף, חיפוש נקודות בתלת ממד עלול לגרום לבא כך שניצור אותה קרן פעמיים]

לאחר מכן כדי שנוכל למצוא נקודות בתוך המעגל נמצא שני וקטורים אורתוגונליים המוכלים במישור (ממש כמו שהגדרנו ב VIEW PLANE). מכיוון שיש לנו את הנורמל למישור (וקטור כיוון האור, ב SPOT LIGHT הוא קבוע וב POINT LIGHT בהתאם לנקודה המוארת) מצאנו וקטור מאונך אליו לפי החישוב שמכפלה סקלרית של הווקטור בנורמל שווה ל 0. ווקטור שני נמצא ע"י מכפלה ווקטורית של הווקטורים הנ"ל.

על ידי שני הווקטורים במישור נוכל לזוז במישור, בתוך שטח המעגל (לפי בדיקות שהגדרנו), ולקבל כל פעם נקודה אחרת בתוך העיגול שמייצג את אזור מקור האור.

להלן הקוד למציאת נקודות במעגל לאחר שיצרנו שתי וקטורי כיוון:

```
Point3D xPoint;    // movement in the x axis
Point3D yPoint;    // movement in the y axis
//number of moves from the center to each side in the direction of vector X(or it opposite direction)
//if the are was a square then the formula was :square (mn Points) / 2
//since the are is a circle (area is about 80 percent) and our coverage is about 80-90
// total we get about 2/3 of the original wanted amount
// there we multiple the previews formula by 1.25 times (square is 1.5 bigger. after 2/3 reduce we
// will get the wanted amount)
int PARTITION = (int) (Math.sqrt(minPoints) / 2 * 1.25);
double distance = _radius / PARTITION;
for (int i = -PARTITION; i <= PARTITION; i++) {
    if (Util.alignZero( number: i * distance) != 0) {
        xPoint = _position.add(x.scale(i * distance));
    } else {
        xPoint = _position;
    }
    double maxY = Math.sqrt((_radius * _radius) - (i * distance) * (i * distance));
    int moves = (int) (maxY / distance);
    for (int j = -moves; j <= moves; j++) {
        if (Util.alignZero( number: j * distance) != 0) {
            pointsInCircle.add(xPoint.add(y.scale(j * distance)));
        }
    }
}
return pointsInCircle;
```

כעת כל שנותר לעשות הוא לשנות את פונקציית transparency, כך שבמקום שתייצר קרן אחת למרכז מקור האור ותבדוק את ההצללה עליו, עכשיו היא תייצר קרניים למספר רב של נקודות שמצאנו קודם ותבדוק את ההצללה בכלם ותחזיר ממוצע משוקלל של כל הנקודות שהוא ההצללה האמיתית בנקודה.

סצנה למיני פרויקט 1:

יצירת סצנה של חדר המורכב מ6 משטחים (מישורים). בתוך החדר שולחן המורכב מ22 גאומטריות. ניתן לראות בתמונה את הצל של השולחן לפני ואחרי השיפור.

על השולחן מונחת פירמידה ובתוכה עוד פירמידה המורכבות כל אחת מ4 משולשים. הפירמידה החיצונית שקופה ולכן ניתן לראות דרכה את הפירמידה הפנימית. (המחשת refraction, transparency)

ארון המורכב מ14 גאומטריות שני הדלתות השמאליות הינם בעלי מקדם השתקפות 1 ולכן יוצרות מראה מושלמת דרכה אפשר לראות את השולחן והפירמידות. (המחשת reflection)

שלושה כדורים על הקיר הימני בהם ניתן לראות את השפעת האור מבחינה ספקולרית ודיפוזיבית.

כמו כן בסצנה שלושה מקורות אור:

- Spot light שנמצא מעל מרכז השולחן וכיוונו מטה אל השולחן.
- Spot light מכוון אל הכדור האמצעי באלכסון ימינה מטה.
- Point light מול הארון שניתן לראות את החזר האור הספקולרי בדלתות הארון הימניות, ואת החזר האור הדיפוזיבי ממנו בתקרה שמעליו.

ניתן לראות את ההבדלים לאחר השיפור SOFT SHADOW בקלות בצל שנוצר על ידי השולחן והארון.

תמונה לאחר השיפור:



תמונה לפני השיפור:

