מיניפ בהנדסת תוכנה :

מגישים: נתנאל נחמן חסיד אורי מגד



מיניפ 1:

שיפור super sampling :

**Super Sampling Anti-Aliasing** בקיצור SSAA הוא אחת מטכניקות החלקת הקצוות (Anti-Aliasing) הראשונות והאיכותיות ביותר בתחום עיבוד התמונה הגרפית, בעיקר במשחקי מחשב ובגרפיקה תלת־ממדית. מטרת הטכניקה היא להפחית את תופעת ה"משוננות" (aliasing) – קווים בעלי מראה משונן או לא אחיד, במיוחד בשיפועים או בקצוות אלכסוניים.

**איך זה עובד?**

Super Sampling פועל על ידי רינדור של התמונה ברזולוציה גבוהה יותר מהנדרש (לדוגמה פי 2 או פי 4), ולאחר מכן ביצוע מיזוג (downsampling) חכם של הפיקסלים לרזולוציית היעד. כלומר, במקום לחשב פיקסל אחד לכל נקודה על המסך, Super Sampling מחשב כמה דגימות לכל פיקסל וממוצע ביניהן יוצר את הפלט הסופי.

כך לדוגמה, פיקסל בודד בתמונה עשוי להיווצר מממוצע של ארבע דגימות, שנלקחו מחלקים שונים בתוך אותו אזור. בצורה זו מושגת תוצאה חלקה יותר, עם פחות קפיצות בגרדיאנטים, פחות קצוות משוננים, ושיפור כללי באיכות התמונה.

**יתרונות**

* איכות תמונה גבוהה מאוד – נחשבת לטכניקה המדויקת והנקייה ביותר מבין טכניקות ההחלקה.
* מפחיתה לא רק קצוות משוננים, אלא גם רעש ו"ריצודים" בתנועה (temporal aliasing).

**חסרונות**

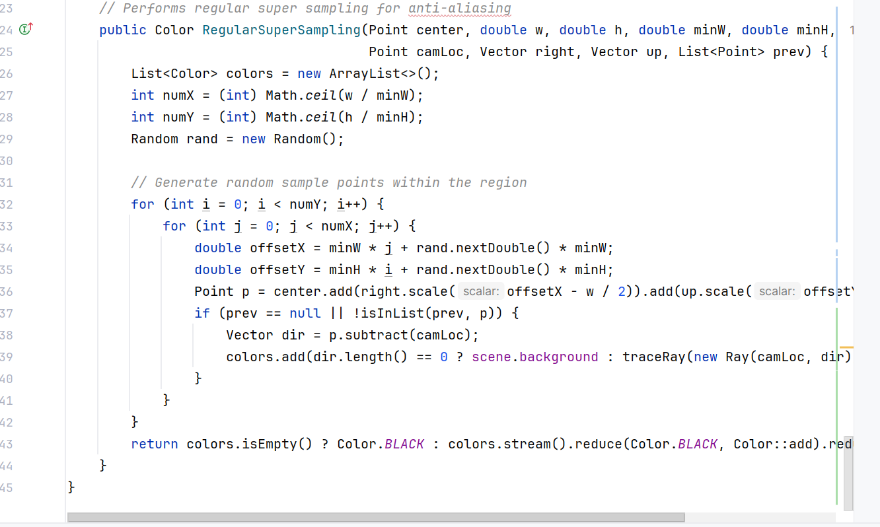
* **עלות חישוב גבוהה מאוד** – כיוון שהיא דורשת רינדור של כל המסך ברזולוציה גבוהה יותר, היא צורכת יותר משאבי עיבוד וכרטיס מסך חזק.
* לא תמיד מעשית במשחקים מודרניים בעלי עומס גרפי כבד, במיוחד ברזולוציות גבוהות.

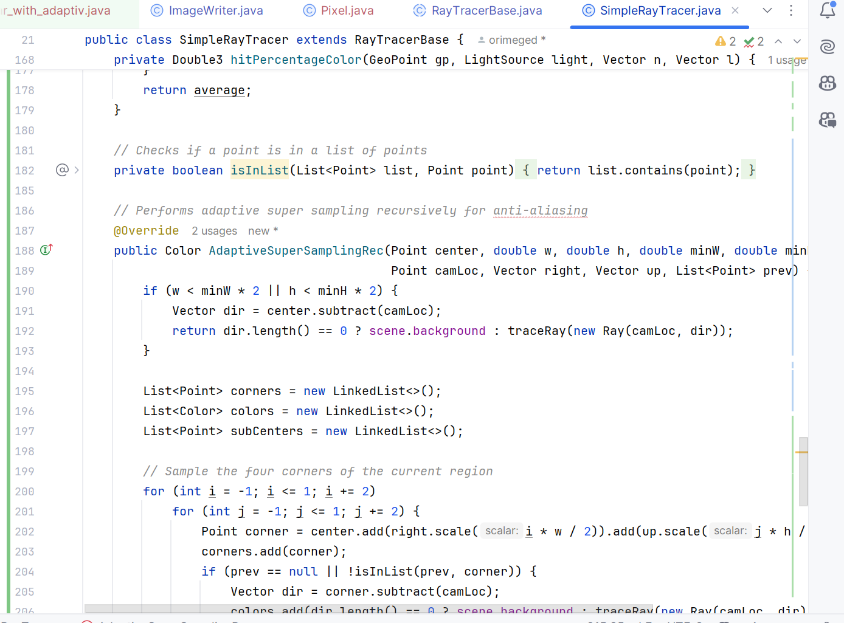
**גרסאות חדשות ומשופרות**

במהלך השנים פותחו גרסאות יעילות יותר, כמו:

* **Multisample Anti-Aliasing (MSAA)** – מבצעת דגימה חכמה רק באזורים הרלוונטיים לקצוות.
* **Temporal Anti-Aliasing (TAA)** – משתמשת במידע מהפריימים הקודמים.
* **NVIDIA DLSS** ו־**AMD FSR** – טכניקות חדשות המשתמשות בלמידת מכונה להפקת תמונה באיכות גבוהה תוך שמירה על ביצועים טובים יותר.

**תמונות מהקוד:**





super sampling:

ללא super sampling:

**מיניפ 2: Adaptive**

**Adaptive Anti-Aliasing** היא שיטת החלקת קצוות (Anti-Aliasing) חכמה, שנועדה לשפר את איכות התמונה על ידי זיהוי והחלקה ממוקדת של אזורים בעייתיים – תוך שמירה על יעילות וביצועים טובים יותר בהשוואה לשיטות מסורתיות כמו Super Sampling.

**הרעיון המרכזי**

בשונה מטכניקות פשוטות שמחלקות את כל המסך באופן אחיד, Adaptive Anti-Aliasing פועלת **באופן דינמי**: היא מזהה את הקצוות שבהם נראית תופעת ה־aliasing בצורה החזקה ביותר – כמו שינויים פתאומיים בצבע או בניגוד – ומבצעת החלקה רק באזורים אלו. שאר התמונה, שבה אין קצוות חדים, נשארת כמעט ללא שינוי.

**איך זה עובד?**

המערכת מבצעת ניתוח של כל פיקסל כדי לבדוק אם הוא נמצא סמוך לשינוי חד בגבול של עצם, קו או טקסטורה. רק אם מתגלה אזור שדורש החלקה, מופעלת עליו טכניקת דגימה מתקדמת, כמו Multisampling או דגימה מרובעת. במילים אחרות – החלקה רק כשצריך.

**יתרונות**

* **שיפור איכות התמונה** במקומות החשובים באמת – הקצוות.
* **שמירה על ביצועים** – נמנעים מיישום החלקה על כל התמונה.
* **התאמה לסצנות דינמיות** – פועלת היטב גם בתנועה או בתמונות מורכבות.

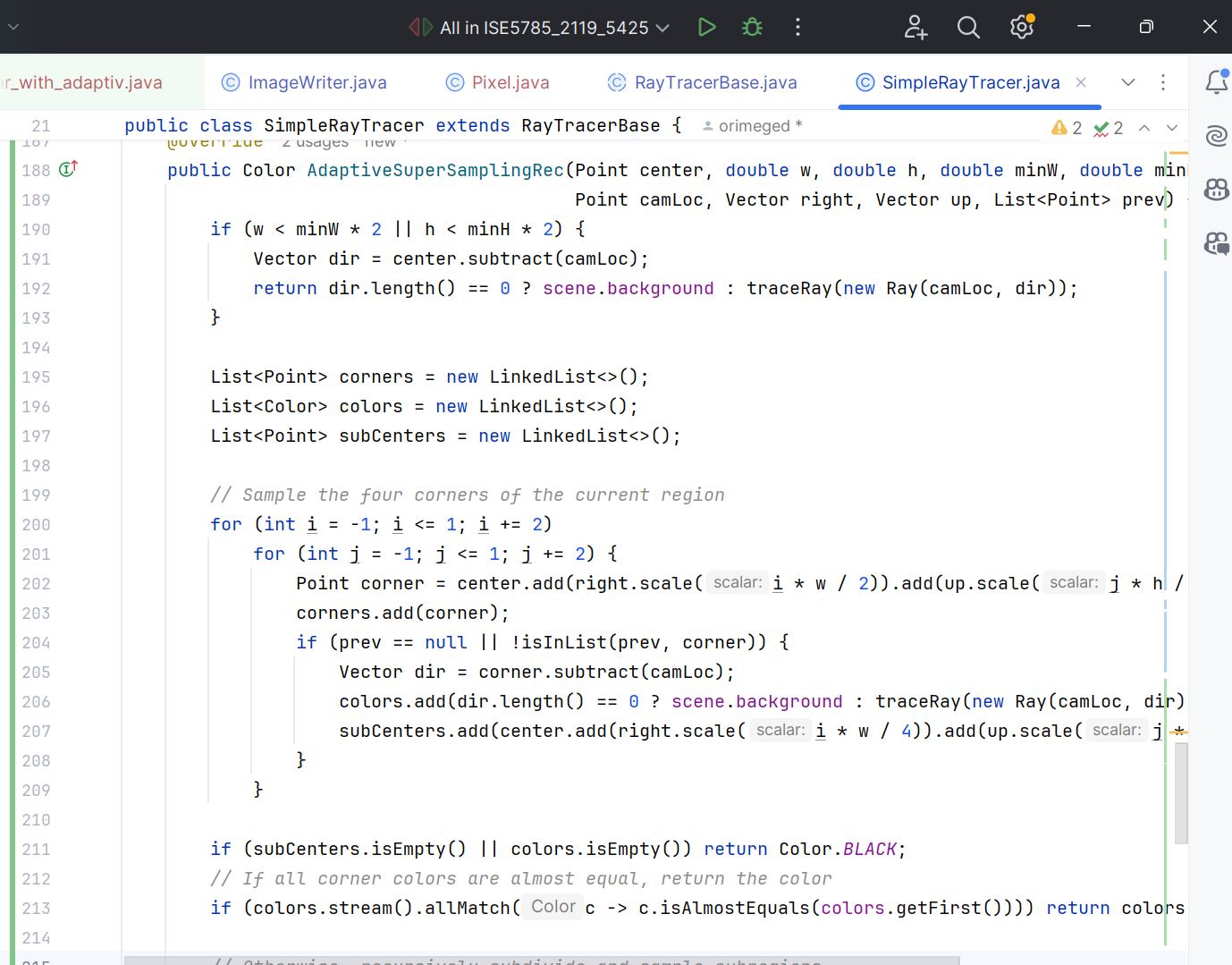
**חסרונות**

* איכות ההחלקה לא תמיד עקבית – במקרים מסוימים יכולים להישאר קצוות חדים.
* תלויה במימוש של הדרייבר או המנוע הגרפי – לא תמיד זמינה בכל משחק או יישום.

**שימושים עיקריים**

Adaptive Anti-Aliasing נפוצה בעיקר בכרטיסי מסך של AMD (בעבר נקראה גם **AAA** – Adaptive Anti-Aliasing) והייתה פתרון יעיל בעידן שבו Super Sampling נחשב כבד מדי. כיום, עם עליית טכניקות מתקדמות כמו DLSS ו־FSR, השימוש בה פחת, אך היא עדיין קיימת כאופציה זמינה במצבים שבהם רוצים איזון בין איכות לביצועים.

**תמונות מהקוד:**





בפרויקט שכלל רינדור של סצנה מורכבת בתלת־ממד ברזולוציה גבוהה, נמצא כי ללא שימוש ב־Adaptive Anti-Aliasing, זמן הריצה הכולל הגיע לכ־שעתיים וחצי. זאת עקב הפעלת החלקת קצוות מסורתית על כל הפיקסלים במסך, גם באזורים שבהם לא נדרשה החלקה בפועל. לעומת זאת, כאשר נעשה שימוש בטכניקת Adaptive Anti-Aliasing, זמן הריצה ירד בצורה דרמטית לכ־20 דקות בלבד. ההבדל נובע מהעובדה ש־Adaptive מבצעת החלקה אך ורק באזורים שבהם יש קצוות חדים או ניגוד חזותי גבוה, וכך חוסכת משאבים יקרים תוך שמירה על איכות ויזואלית מרשימה. מדובר בקיצור של מעל 85% בזמן העיבוד – שיפור משמעותי גם מבחינת יעילות וגם מבחינת ניצול חומרה.