תרגיל בית 5

סימולציה של מערכת מרובת גופים

יש להגיש תרגיל הזה עד סוף הסמסטר, 29 בינואר 2017. (**אל תבנו על הארכה!**) כללו בהגשתכם את קובצי הקוד[1] וצילום מסך של התצוגה. את הכל יש לארוז בפורמט הרגיל, 3_Ex5_*xxxxxxxxx_yyyyyyyyy*.zip, שבו x* ו-y* הם מספרי הזהות של המגישים. את הקובץ הזה יש להעלות לאתר הקורס.

התרגיל הנוכחי מרכז שלושה נושאים:

1. שימוש בגרפיקה עם Swing
2. תבנית העיצוב של Model – View – Control (MVC)
3. קצת פיסיקה ניוטונית ☺

אפשר שזכור לכם משעורי הפיסיקה משהו על ניוטון (1643 – 1727, זה עם התפוח...) וחוק הכבידה, שמסביר את האינטראקציה בין שתי מסות תחת ההשפעה ההדדית של כבידה. אולי גם ידוע לכם שלבעיה עם שני גופים יש פתרון מדוייק וסגור, אבל עד כה לא נמצא פתרון עבור מערכות עם שלושה גופים או יותר. הדרך היחידה ללמוד, לנתח ולחזות התנהגות של מערכות כאלה היא באמצעות **סימולציה**.

סימולציה היא תהלך חישובי בו מדמים (לרוב באמצעות מחשב) את המערכת מרובת הגופים. השיטה המקובלת היא שיטת "חמור ארוך": בוחרים יחידת זמן קצרה יחסית לזמן הכולל שבו רוצים לצפות במערכת. (אנו נסמן גודל זה ב-Δt.) למשל, אם רוצים לבדוק התפתחות במשך 10 שנים, אפשר לבחור יחידה של יום אחד בתור יחידה קצרה. את מצב המערכת מקדמים בקפיצות בגודל יחידת הזמן שנבחרה, תחת הנחת קירוב: כוחות, מהירויות ומיקום אינם משתנים במשך יחידה אחת אלא רק מיחידה אחת לשניה.

הגדרת המשימה

יש לכתוב תכנית Java מבצעת את הדברים הבאים:

1. מקבלת פרמטר מטיפוס double משורת הפקודה (פרמטר ל-()main) שמציין את Δt, יחידת הזמן הקצרה של הסימולציה. (כל פרקי הזמן מצויינים בשניות: ביממה, למשל, יש 86,400 שניות, אז הפרמטר שמציין יממה הוא 8.64e5.)
2. קוראת את הרכב היקום שיש לדמות בסימולציה מקובץ קלט בשם Universe.txt. מבנה הקובץ יתואר בהמשך. קובץ המתאר את השמש וארבעת כוכבי הלכת הקרובים אליה ביותר מצוי באתר.
3. פותחת חלון גרפי על המסך ומציירת בו את הכוכבים שביקום, לפי מיקומם ההתחלתי בקובץ.
4. מבצעת סימולציה של היקום, החל מזמן $t = 0$ בשיטת "חמור ארוך" שמתוארת להלן. אחרי כל שלב בסימולציה, התכנית מעדכנת את תמונת היקום ומקום הכוכבים בו.

[1] אלה עם סיומת java.

תיאור היקום

היקום מתואר בקובץ טקסט המכיל את הפרטים הפיזיקליים של כל הכוכבים שבו.

בשורה הראשונה של הקובץ יש מספר שלם יחיד N, שמציין את מספר הכוכבים שביקום.
בשורה השנייה יש מספר ממשי יחיד המציין את רדיוס היקום R, במטרים. מספר זה יהיה שימושי בקביעת קנה המידה של התצוגה: כמה מטרים מייצג כל פיקסל על המסך.

בהמשך ימצאו N שורות נוספות, אחת לכל כוכב. בכל שורה נמצאים ששת הערכים הבאים:

1-2: מיקום הכוכב: קואורדינטות ה-x וה-y של הכוכב, יחסית לראשית שנמצאת במרכז היקום. (כל המרחקים מצויינים במטרים.)

3-4: מהירות הכוכב: הרכיבים v_x ו-v_y של המהירות $\vec{v}$ של הכוכב (ביחידות של מטר לשניה).

5: מסת הכוכב (בקילוגרמים).

6: תמונת הכוכב: מחרוזת המכילה את שם קובץ ה-jpg של תצלום של הכוכב. קובצי ה-jpg מצויים אף הם באתר, ויש להעתיקם לאותה תיקיה בה נמצא קובץ תיאור היקום. גודל כל תמונה יחסי לגודל הפיזי של הכוכב המתואר בה, פרט לזה של השמש, שתמונתה הרבה יותר קטנה מגודלה יחסית לכוכבי הלכת שמסביבה.

להלן תוכן קובץ תאור היקום שמצורף לתרגיל זה. הוא מתאר את השמש ואת ארבעת כוכבי הלכת הקרובים אליה ביותר: כוכב החמה (מרקיורי), נגה (ונוס), ארץ ומאדים (מארס).

```
5
2.50e+11
 0.0000e+00  0.0000e+00  0.0000e+00  0.0000e+00  1.9890e+30      Sun.jpg
 5.7900e+10  0.0000e+00  0.0000e+00  4.7900e+04  3.3020e+23  Mercury.jpg
 1.0820e+11  0.0000e+00  0.0000e+00  3.5000e+04  4.8690e+24    Venus.jpg
 1.4960e+11  0.0000e+00  0.0000e+00  2.9800e+04  5.9740e+24    Earth.jpg
 2.2790e+11  0.0000e+00  0.0000e+00  2.4100e+04  6.4190e+23     Mars.jpg
```

כפי שניתן לראות, השמש נמצאת במרכז היקום שלנו ומהירותה אפס; כוכבי הלכת מצויים כולם על ציר ה-x ולמהירותם ההתחלתית יש רכיב בכוון y בלבד. (יתכן שהיה רגע בחמשת ביליוני השנים האחרונות בהם כוכבי לכת סובבים את השמש שארבעת אלה שכאן אכן ניצבו בקו ישר אחד, מי יודע...)

הפיזיקה

נעבור על הפיזיקה שמגדירה את תנועת גרמי השמיים לפי חוקי ניוטון. אל תחששו אם המכניקה שלכם מעט חלודה: כל הנוסחאות הרלוונטיות נמצאות כאן...

נניח לרגע שהמיקום (x, y) והמהירות (v_x, v_y) של כל כוכב ביקום ידועים. כדי לחשב איך היקום משתנה, עלינו לדעת מהו הכח הפועל על כל אחד מהכוכבים. שלושה עקרונות ניוטוניים משחקים כאן:

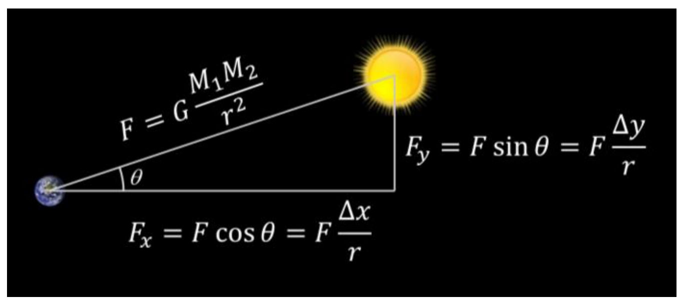
<u>הכוח הפועל בין שני כוכבים:</u> לפי חוק הכבידה של ניוטון, שתי מסות מושכות זו את זו בכוח שעומד ביחס ישר למכפלת המסות שלהם וביחס הפוך לריבוע המרחק. אם נסמן את מסות הכוכבים ב-M_1 ו-M_2 ואת המרחק ביניהם ב-r, עצמת המשיכה ביניהם F היא

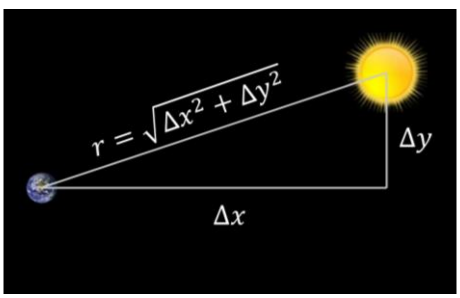
[נוסחה 1]
$$F = G\frac{M_1 M_2}{r^2}$$

ערכו של קבוע היחס, המסומן לרוב ב-G, הוא

[נוסחה 2]
$$G = 6.67\times10^{-11}\,\frac{Nm^2}{kg^2}$$

כיון שאנו משתמשים במערכת צירים קרטזית, נוח לפרק גם את הכוח לרכיביו, F_x ו-F_y.





<u>הכוח הכולל הפועל על כוכב:</u> על פי עקרון הסופרפוזיציה, הכח הכולל הפועל על כוכב הוא סכום כל הכוחות הפועלים עליו. יתר על כן, רכיב ה-x של הכוח הכולל שווה לסכום רכיבי ה-x של כל הכוחות הפועלים עליו, וכן לגבי רכיב ה-y. אפשר אם כן לחשב את האינטראקציה בין כוכב לכל אחד מהכוכבים האחרים, ולסכם את הרכיבים של כל אחת מהן על מנת לקבל את רכיבי הכוח הכולל.

<u>תאוצה:</u> לפי החוק השני של ניוטון, התאוצה של גוף נמצאת ביחס ישר לכוח הפועל עליו וביחס הפוך למסה שלו. אם נפרק את התאוצה לרכיביה, נקבל:

[נוסחה 3]
$$a_x = \frac{F_x}{M} \qquad a_y = \frac{F_y}{M}$$

הסימולציה

בשיטת ״חמור ארוך״ אנו מבצעים אינטגרציה נומרית של המשוואות שלעיל. מחלקים את הזמן הרציף ליחידות (יחסית קטנות) באורך Δt, ומחשבים את מצב המערכת תחת ההנחות הבאות:

1. זמן מתקדם ביחידות של Δt.
2. במשך הזמן הקצר Δt, המקום של הכוכב משתנה כאילו לא פועל עליו שום כוח.
3. כמו כן, במשך הזמן הקצר Δt המהירות של הכוכב משתנה כאילו אינו משנה את מקומו.

הנחות אלה מאפשרות את החישוב: בלעדיהן, הכוח שפועל על הכוכב משתנה תוך כדי תנועתו, ואין פתרון אנליטי לבעיה כאשר יש ביקום יותר משני כוכבים. שיטת חישוב זו משמשת באנליזות של תהליכים קוסמולוגיים ובחישוב מסלולי לווינים, למשל.

הצעדים המתוארים להלן מסבירים כיצד משתנים המקום והמהירות של כל כוכב בכל שלב של הסימולציה:

- **צעד 1** (מחושב בנפרד עבור כל כוכב): מחשבים את הכוח שפועל על הכוכב בזמן t עקב האינטראקציה שלו עם כל אחד מהכוכבים האחרים. מסכמים את רכיבי ה-x וה-y של כל אחד מהם כדי לקבל את רכיבי הכוח הכולל (F_x, F_y). זכרו שמקום, מהירות, תאוצה וכוח הם כולם ווקטורים, ושלרכיבים שלהם יש סימן שחשוב לשמור ולהתייחס אליו. למשל, בציור שלעיל השמש מושכת את הארץ כלפי מעלה וימינה, כך שהרכיבים של הכוח שפועל בגינה על הארץ גדולים מאפס שניהם, אך המצב ישתנה כאשר כדור הארץ בהקיפו את השמש ימצא בצידה האחר.
- **צעד 2** (מחושב עבור כל כוכב אף הוא):
  - מחשבים את התאוצה (a_x, a_y) בזמן t על פי $a_x = \frac{F_x}{M} \qquad a_y = \frac{F_y}{M}$
  - מחשבים את מהירותו החדשה של הכוכב (לזמן $t + \Delta t$): $v_x + \Delta t \times a_x \qquad v_y + \Delta t \times a_y$
  - מחשבים את מיקומו החדש של הכוכב (לזמן $t + \Delta t$). משתמשים לצורך החישוב בערכי המהירות שחושבו בצעד הקודם: $x + \Delta t \times v_x \qquad y + \Delta t \times v_y$
- **צעד 3**: מעדכנים את מיקומו של הכוכב על המסך על פי הקואורדינטות החדשות שלו.

הגרפיקה

יצירת תמונת המסך היא ישירה: יש להגדיר מחלקה שיורשת מ-JPanel ודורסת את מתודת ה-paintComponent() שלה. במתודה הדורסת יש למחוק תחילה את התמונה (על ידי ציור מלבן מלא ושחור על פני כל החלון) ואחר כך לצייר את כל אחד מהכוכבים על פי מקומו הנוכחי (באמצעות המתודה drawImage() של Graphics). אם המתודה נקראת 30-40 פעם בשניה מתקבל הרושם של תנועה רציפה. בדקו בתיעוד של מחלקות אלו פרטים שאולי חסרים לכם.

ארגון הקוד

אקבל כל תכנית שמבצעת את המתואר לעיל כנכונה. אלא שליישומים מהסוג הזה, שבו יש אלגוריתמים שרצים ברקע ומשפיעים על תצוגה שמראה את מצב המערכת באופן שוטף, יש תבנית עיצוב מקובלת שמקלה מאד על ארגון הקוד, ואשמח אם תנסו להשתמש בה כאן.

תבנית זו נקראת Model-View-Control או בקיצור MVC. תבנית ה-MVC מחלקת את האחריות להפעלת היישום בין שלושה גורמים:

1. **Model**: חלק זה אחראי על הייצוג הפנימי של מצב המערכת. בתרגיל זה, כאן תשמר רשימה של הכוכבים וכל התכונות הרלוונטיות שלהם: מסה, מיקום ומהירות. המודל מתפתח תוך כדי חיי התכנית, והערכים השמורים בו מתעדכנים עם הזמן בידי ה-Control.

2. **View**: חלק זה אחראי על התצוגה. תפקידו לחקור את ה-Model ולהפיק ממנו את המידע שנחוץ על מנת לייצג את מצב המערכת. העיבוד שנחוץ על מנת להפוך את הייצוג לנוח למשתמש מקומו בחלק הזה. כמו כן, חלק זה אחראי לקבל הוראות וקלטים מהמשתמש ולדווח על כך ל-Control. (צד זה של האחריות של View אינו מתבטא בתרגיל שלפננו, כיון שאין לתצוגה שלנו אינטראקציה עם מפעיל.) בתרגיל זה, View הוא מועמד טוב למחלקה שצריכה לממש את paintComponent().

3. **Control**: כאן נמצאת הלוגיקה של התכנית. חלק זה כולל את האלגוריתמים שמנהלים את היישום. לכאן מגיעות ההוראות של המפעיל (שמגיעות דרך ה-View) וכאן מחליטים כיצד לפעול כדי למלא אחריהן. תוצאות הפעולות נרשמות ב-Model. בתרגיל זה, חישובי הסימולציה מקומם בחלק הזה.

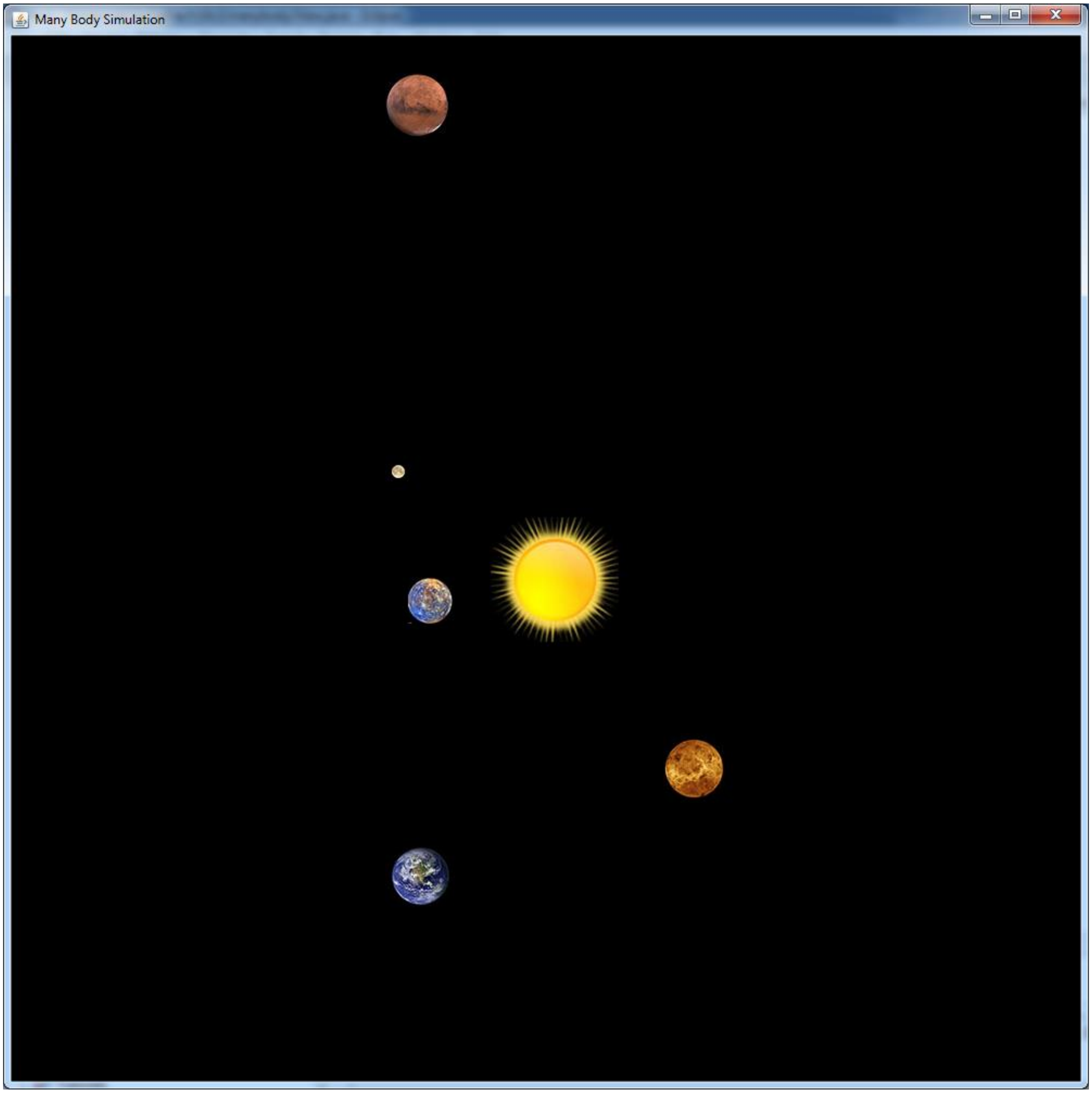
מטרת ההפרדה הנ"ל היא לאפשר מודולריות טובה ביישום. למשל, יישום שפותח למסך של מחשב יוכל להיות מותאם למסך קטן של טלפון נייד על ידי שנויים ב-View בלבד, וללא השפעה על החלקים האחרים של היישום. בדומה, שנויים באלגוריתם ירוכזו רק בחלק של ה-Control. שימו לב שה-Control לא מדווח למשתמש על פעולותיו: במקום זאת הוא מעדכן את ה-Model, ושינויים אלה מוצאים את דרכם לתצוגה מעצמם, עקב עבודת הדווח השוטף שמבצע ה-View.

הרשת שופעת הסברים על תבנית זו, שיחד עם תבנית ה-Observer (מה שב-Swing נקרא Listener) מהוות את התבניות המקובלות והשימושיות ביותר.

כרגיל, שאלות ותהיות יש להעלות לפורום, ואני אשתדל להגיב להן במהירות.

בהצלחה!

כדי לסבר את העין, צירפתי כאן תמונה שנוצרה על ידי הפתרון שלי לתרגיל.



יש כאן גרם שמימי נוסף שמדמה כוכב שביט – אתם מוזמנים לעשות ניסויים שכאלה.

השביט הוא מאד קטן, מהירותו קטנה ומרחקו מהשמש מאד גדול. כתוצאה מכך הוא נמשך כמעט בדיוק לכוון השמש, אבל הוא אינו פוגע בה אלא מקיף אותה במהירות רבה וחוזר למרחקים הגדולים מהם הגיע, גם במציאות וגם בסימולציה, למרבה הפלא...