

מטלה 2 – תכנות מתקדם בשפת ++C

הנחיות: קראו היטב את ההנחיות, אי עמידה בהנחיות אלו תגרור הורדת ציון.

- כל שאלה על המטלה נא להפנות לאחראי התרגיל: גיל לוי.
- העבודה וההגשה ביחידים בלבד!
- תאריך הגשה: 30.12.17
- יש להגיש למערכת ה moodlearn עד מועד ההגשה, אם יש סיבה מוצדקת לאיחור בהגשה יש לדווח עד יומיים לפני מועד ההגשה, לאחר מכן לא תתקבל שום סיבה לאיחור.
- ניתן להגיש באיחור של עד 3 ימים תמורת 10 נקודות ובאיחור של עד שבוע תמורת 15 נקודות.
- **לא ניתן להגיש באיחור של יותר משבוע ללא סיבה מוצדקת.**
- יש להגיש את כל הקבצים בתיקייה המכילה את קבצי הפרויקט `cpp`. וקבצי `h`. ובנוסף יש להגיש קובץ `cmake` ואת קובץ ה `makefile` שנוצר, את התיקייה יש לכווץ לקובץ ששמו הוא ת"ז של הסטודנט.
- חובה לכתוב בקובץ `CmakeLists.txt` עבור כל פקודה תיעוד מפורט מה הפקודה עושה.
- שימו לב שחובה שהפרויקט יעבור קומפילציה בקומפיילר `g++`. ניתן להשתמש גם ב `C++11`
- שימו לב לתכנות נכון ולפי כל הכללים של "תכנות מונחה עצמים" שנלמדו בכיתה. בהחלט מותר להוסיף פונקציות או מחלקות נוספות מעבר למה שאתם נדרשים בתרגיל ולהגדיר את הקשרים ביניהם לצורך תכנות נכון בפרט יש לשים לב:
 - מתי יש להשתמש בירושה ובפולימורפיזם.
 - אלו פונקציות יהיו וירטואליות.
 - מתי להשתמש במחלקות אבסטרקטיות.
 - לשמור על עיקרון הכימוס.
 - לשים `const` במקומות שצריך.
 - מתי להשתמש במצביע, מתי ברפרנס ומתי בערך רגיל.
 - חובה לתעד כל פונקציה שאתם כותבים.
 - יש להימנע ממספרי קסם.
 - חובה עליכם לבצע בדיקות שאין לקוד שלכם דליפת זיכרון.
- לפני שאתם מתחילים לעבוד על התרגיל מומלץ לעבור ולהבין את כל המטלה כדי שתוכלו לכתוב ולעצב את הקוד בצורה הנכונה ביותר.

הנחיות נוספות למטלה

- חובה עליכם להשתמש ב `cmake` כדי ליצור קובץ `makefile` – הגישו גם את קובץ ה `CmakeLists` וגם את קובץ ה `makefile` שנוצר.
- עליכם לכתוב `class diagram` המתאר את כל הפרויקט שלכם.
- עליכם לצרף קובץ טקסט המתאר את השימוש שלכם בעקרונות תכנות מונחה עצמים ואת המקומות בהן השתמשתם ב `design patterns`.
- יש לבדוק תקינות כל קלט ואין להניח קלט תקין. בנוסף, כל אילוף חייב להיבדק.

המטלה – משחק מלחמה

תיאור כללי:

בתרגיל זה עליכם לכתוב משחק מלחמה – משחק שבו יש מספר צבאות הנלחמים בתוך שדה קרב מדומה.

- מספר השחקנים במשחק הוא לכל היותר 4 שחקנים, מינימום – 2 שחקנים.
- כל שחקן יכול להיות מוגדר כשחקן אנושי או שחקן ממוחשב.
- לכל שחקן, ישנו צבא של n חיילים העומדים לרשותו כאשר מספר החיילים הוא זהה לכל הצבאות, בכל צבא, החיילים מתחלקים למספר סוגי חיילים (סוגי החיילים כן יכולים להשתנות מצבא לצבא), כפי שיפורט בהמשך.
- בנוסף לחיילים, הזירה תכיל פריטים הניתנים לאיסוף ואובייקטים שונים כפי שיוגדר בהמשך.
- בכל תור משחק שחקן אחר ומחליט להזיז את אחד החיילים שלו למיקום חדש, ואז לתקוף חייל של שחקן אחר וכו'.
- המשחק מתחיל בכל פעם בתצורה (קונפיגורציה) התחלתית, אותה יש לקרוא מתוך קובץ. פורמט הקובץ יוגדר בהמשך. בקובץ זה נגדיר את מספר השחקנים והסוג שלהם (מחשב או אנושי), גודל זירת הקרב, מספר החיילים לכל שחקן, הסוג שלהם, המיקום ההתחלתי, הנשקים שעומדים לרשותם וכו'.
- הנתונים הללו ייקראו בתחילת המשחק ועל פיהם יאותחלו כל האובייקטים השונים.
- ממשק הקלט למשחק – הצעדים של השחקנים. וממשק הפלט – תצוגת הזירה וכו' יפורטו בהמשך.
- בסיום המשחק על התוכנית שלכם לייצר קובץ המכיל את כל הקונפיגורציות של המשחק (מספר החיילים לכל שחקן ומיקום החיילים בכל איטרציה של המשחק, מיקום הפריטים וכו') ובנוסף עליכם להוסיף בסוף קובץ זה את המנצח במשחק או לציין שהמשחק לא נגמר (יוגדר בהמשך).

מימוש התרגיל

בתרגיל זה תקבלו **חלק** מהמחלקות שעליכם לכתוב עם המאפיינים – יש לעצב את התוכנית באופן הנכון ולהוסיף פונקציות או מחלקות במידת הצורך. בהחלט יש צורך להוסיף `getters/setters` במידת הצורך. חובה להוסיף לכל מחלקה פונקציית `toString` לתיאור האובייקט.

➤ המחלקות במשחק

1. מיקום – Point2d

מחלקה זו מייצגת נקודה במישור ובה נשתמש לייצג את מיקום השחקנים/הפריטים. ערכי הנקודה יהיו מספרים מסוג `double`.

2. שחקן – Player

מחלקה זו מייצגת שחקן במשחק (שחקן אנושי או שחקן מחשב), לכל שחקן יהיו n חיילים העומדים לרשותו כאשר הפרמטר n וסוג השחקן מתקבלים מקובץ הקונפיגורציה.

אם השחקן הוא שחקן מחשב, יועבר בנוסף (בקובץ) פרמטר (מספר) המציין את האסטרטגיה בה ישתמש המחשב מתוך רשימת האסטרטגיות הקיימות בתוכנית. למשל: הגרלה של המיקום החדש אליו מזיזים את השחקן בכל שלב.

אין צורך לחשוב על מספר אסטרטגיות חכמות. עליכם רק לממש אסטרטגיה אחת בה המחשב מתקדם עם החיילים שלו באופן אקראי ויוגדר לו רדיוס (באסטרטגיה) שממנו הוא מתחיל להתקיף וכל עוד ניתן להתקיף – המחשב תוקף.

3. חייל - Soldier

הצבא יהיה מורכב מסוגי החיילים הבאים, אך אין חובה להשתמש בכולם וניתן להרכיב צבא מכל קומבינציה אפשרית.

סוגי החיילים:

- חייל רגיל – יש לו את התכונות הבאות:
 - א. כמות חיים התחלתית: 100 יחידות.
 - ב. מהירות הליכה – בכל איטרציה של המשחק הוא יכול לנוע עד רדיוס 2 ממקומו הקודם.
 - ג. יכולת התקפה – סיכוי הפגיעה שלו בחייל אחר העומד במרחק d ממנו $= 1/d$.
 - ד. בנוסף, יש לחייל רגיל יכולת ריצה שבה הוא יכול לנוע עד רדיוס 4 ממקומו הקודם, במידה והפרמטר שהועבר למתודת ההליכה גדול לפחות פי 4 מהרדיוס המקסימלי אותו הוא יכול ללכת. המחיר לריצה הוא ירידה של 10 יחידות חיים.
- חייל צלף – יש לו את התכונות הבאות:
 - א. כמות חיים התחלתית: 80 יחידות.
 - ב. מהירות הליכה – בכל איטרציה של המשחק הוא יכול לנוע עד רדיוס 2 ממקומו הקודם.
 - ג. יכולת התקפה – סיכוי הפגיעה שלו בחייל אחר העומד במרחק d ממנו $= (d - 1)/d$.
- חייל חובש – יש לו את התכונות הבאות:
 - א. כמות חיים התחלתית: 120.
 - ב. מהירות הליכה – בכל איטרציה של המשחק הוא יכול לנוע עד רדיוס 4 ממקומו הקודם.
 - ג. יכולת התקפה – חייל זה לא נושא נשק כלל ולכן ההתקפה שלו היא רק עם ידיו – הנזק שיעשה לחייל המותקף הוא הורדה של 10 יחידות חיים.
 - ד. לחייל יש פעולה לרפא חיילים אחרים (שהם מהצבא שלו) כאשר הוא נמצא ברדיוס של עד 4 מהם. הפעולה תקבל את החייל שאותו יש לרפא. כמות החיים של החייל המתרפא תחזור למקסימום שלו.

4. פעולות שחיילים מבצעים

הפעולות הבסיסיות שחיילים מבצעים במשחק:

- הליכה – פעולה זו תקבל נקודת יעד והחייל יצעד לכיוון של הנקודה עד המרחק שהוא מסוגל להגיע. אם כמות החיים של החייל קטנה ממש מ-30 – מהירות החייל תהיה פי 2 פחות.
- התקפה – פעולה זו תקבל נקודת יעד להתקפה. החייל ישתמש בנשק שבידו (במשחק זה נניח שמספר הכדורים בנשק הוא אינסופי).

כאשר כמות החיים של חייל יורדת ל 0 (או מתחת ל 0) – החייל מת, לא מבצע שום פעולה נוספת ולא נמצא יותר בזירה כמו גם נשקו.

בקובץ הקונפיגורציה של כל שחקן יהיה רצף של תנועות של חייל אנושי. חייל ממוחשב יחשב בעצמו את הצעד הבא. לכל חייל נקרא בתחילת התוכנית את הרצף שלו, כך שבכל איטרציה הוא ידע לאן עליו ללכת.

לאחר כל הליכה מתאימה, הוא יברר האם יש חיילים בסביבתו, ויבחר את מי לתקוף (שיקול שלכם – הוסיפו את הדרך שבחרתם לקובץ הטקסט).

אם מדובר בחייל חובש, אזי בנוסף הוא גם יברר אם יש בסביבתו, אחרי שצעד, את מי לרפא.

נסו לחשוב כיצד לוודא שחייל לא ייתקוף חיילים מהצבא שלו. וכן שחובש לא ירפא חיילים של היריבים.

פעולת ההתקפה עצמה תתבצע במחלקת Game והיא תקבל שני פרמטרים: החייל התוקף והחייל המותקף. מחלקת Game גם תספק ממשק לקבלת רשימת כל החיילים שבסביבת החייל השואל, וממשק לשאלה האם החייל הנבדק הוא מצבא זר או לא. מכיוון שמחלקת Game אחראית על עוד נושאים, השתמשו בחומר שנלמד בהרצאה על נושא ה-ISP כדי לממש זאת בצורה הנכונה.

5. פריטים במשחק

הפריטים ימוקמו בזירת הקרב בתחילת המשחק (המיקום וסוגי האובייקטים יהיו מתוך הקובץ ההתחלתי).

הפריטים מתחלקים ל-2 סוגים – אובייקטים הניתנים לאיסוף ואובייקטים שאינם ניתנים לאיסוף. כאשר חייל מתקרב עד לרדיוס של 2 מפריט הניתן לאיסוף – הוא לוקח אותו. אם הפריט הוא נשק שאין כרגע לחייל – החייל יחליף את נשקו הנוכחי בנשק החדש וישאיר במקומו את הנשק שלו. שימו לב: חייל יכול להחזיק בכל רגע נתון רק נשק אחד.

א. נשקים: (ניתן לאיסוף)

לכל נשק במשחק יש את המאפיין של עוצמת הנשק ומיקום הנשק. הנשקים במשחק מתחלקים ל 2 סוגים: נשק כבד ונשק רגיל.

הנשקים במשחק זה יהיו:

נשקים רגילים:

- M16 – עוצמת הנשק: 50.
בכל איטרציה של המשחק, ניתן לירות בנשק זה כדור אחד.
- UZI – עוצמת הנשק: 40.
בכל איטרציה של המשחק, ניתן לירות בנשק זה 3 כדורים – כלומר, הסיכוי לפגיעה גדול יותר.

נשקים כבדים:

- Missile – עוצמת הנשק: 100.
בכל איטרציה של המשחק, ניתן לירות בנשק זה כדור אחד.

שימו לב שכמות הכדורים אינה מוגבלת.

ב. הגנות: (ניתן לאיסוף)

לכל הגנה יש את המאפיינים הבאים: רמת הגנה ומיקום ההגנה.

ישנן 2 סוגי הגנות:

- הגנת גוף – BodyArmor – נגד נשקים רגילים
- מגן – ShieldArmor – נגד נשקים רגילים ונשקים כבדים

החייל יכול להשתמש בשתייהן בו זמנית.
הערכים שרמת ההגנה יכולה לקבל הם בין 0.65 ל 1.0 כאשר הפגיעה בשחקן תחושב כמכפלת עוצמת הפגיעה בערך ההגנה ולכן 1.0 – לא מגן כלל ו 0.65 זו ההגנה הטובה ביותר.

ג. אובייקטים מוצקים:

אלו אובייקטים שלא ניתן לעבור דרכם ולא ניתן לירות דרכם.

לכל אובייקט כזה יהיה אורך ורוחב האובייקט כאשר האורך והרוחב המינימאליים הם 2 יחידות. ממשו אובייקט אחד כזה לבחירתכם.

6. מחלקת Game

זוהי המחלקה הראשית, בה יש לקרוא את כל הנתונים מהקבצים, לטפל בהרצת המשחק ולכתוב את התוצאות. שימו לב – עליכם לעצב את חלוקת המשימות כאן ולהחליט האם להשתמש במחלקות נוספות.

מחלקה זו תכיל את זירת המשחק – אובייקט המכיל את כל הנתונים על הזירה בכל רגע.

7. ממשק קלט עבור שחקן

כתבו ממשק לקלט עבור שחקן אנושי – חישבו מה לשים בממשק זה.

ממשו מחלקה בשם FileControl – המקבלת קובץ csv המתאר את סדרת הפעולות עבור שחקן אחד ובו בכל שורה תהיה נקודת יעד עבור השחקן לפי הפורמט המצורף לתרגיל.

שימו לב שנרצה להשתמש בשיטות קלט שונות – חישבו כיצד לעשות זאת נכון.

8. ממשק פלט

כתבו ממשק עבור פלט ובו יהיו 2 פעולות (ניתן להוסיף):

- קבלת פלט עבור מצב נוכחי של זירת המשחק (התאימו לפורמט המצורף) ע"י האופרטור <<.
- קבלת פלט המתאר את סוף המשחק.

ממשו מחלקה הדואגת לייצר בסוף המשחק קובץ csv ובו בכל שורה תהיה קונפיגורציה של המשחק (לפי פורמט המצורף לתרגיל). ובנוסף, בשורה האחרונה יופיע הפלט המתאר את סוף המשחק.

הרצת המשחק

התוכנית צריכה לקבל ארגומנטים באופן הבא:

```
./program_name [initFile] [player1 player2 ...]
```

כאשר:

program_name = שם קובץ הריצה שיצרתם.

initFile = שם קובץ הקונפיגורציה ההתחלתית.

file1 file2... - קבצי הפעולות של השחקנים – אם צריך, לפי סדר השחקנים האנושיים שהוגדר.

דוגמא הפעלה:

```
./EX1 init.csv p1.csv
```

ניתן לקבל מידע על אופן הפעלת התוכנית ע"י הדגל help, לדוגמא:

./ Ex1 -help

שידפיס את השורה הבאה (בפורמט הזה):

[initFile] [player1 player2 ...]

אם יש שגיאה בקובץ האתחול – על התוכנית להסתיים ולהדפיס הודעת שגיאה.

לאחר האתחול, המשחק יתחיל לרוץ באיטרציות (לולאת המשחק) – בכל איטרציה, כל השחקים יבצעו פעולה אחת כאשר שחקן מחשב יבצע פעולה אחת לפי האסטרטגיה שלו וכל שחקן יבצע פעולה אחת לפי הפעולה הבאה המוגדרת לו בקובץ/בממשק קלט אחר. אם קיימת פעולה לא תקינה בקובץ – יש לדלג לפעולה הבאה.

לאחר כל איטרציה, על התוכנית לעדכן את הזירה למצב הנוכחי לאחר כל השינויים.

אם הסתיימו כל הפעולות והמשחק לא נגמר – יש לסיים את המשחק במצב הנוכחי שלו.

פורמט הקבצים – קבצי csv

קובץ הקונפיגורציה ההתחלתית יהיה בפורמט: (ראו קובץ מצורף)

Game

[רוחב זירת הקרב], [אורך זירת הקרב], battlefieldSize

[מספר השחקנים], players

[מספר החיילים לכל שחקן], soldiers

<מספר אסטרטגיה – עבור שחקן מחשב > [האם השחקן אנושי או מחשב], p1

[נשק התחלתי (אם יש)], [מיקום התחלתי], [הסוג של חייל 1]

[נשק התחלתי (אם יש)], [מיקום התחלתי], [הסוג של חייל 2]

....

[האם השחקן אנושי או מחשב], p2

[נשק התחלתי (אם יש)], [מיקום התחלתי], [הסוג של חייל 1]

[נשק התחלתי (אם יש)], [מיקום התחלתי], [הסוג של חייל 2]

....

Objects

<ארגומנטים>, [מיקום האובייקט], [שם האובייקט], [סוג האובייקט: נשק/הגנה/אובייקט מוצק]

<ארגומנטים>, [מיקום האובייקט], [שם האובייקט], [סוג האובייקט: נשק/הגנה/אובייקט מוצק]

....

כאשר:

אם זו הגנה – הארגומנט יהיה רמת ההגנה.

אם זה אובייקט מוצק – הארגומנטים יהיו אורך ורוחב האובייקט.

כל נקודה המייצגת מיקום תהיה מהצורה $[x, y]$ (ללא פסיק)

קובץ הקונפיגורציה לכל שחקן אנושי יהיה בפורמט:

Player

..., [נקודת יעד 3], [נקודת יעד 2], [נקודת יעד 1 (עבור חייל 1)]

..., [נקודת יעד 3], [נקודת יעד 2], [נקודת יעד 1 (עבור חייל 2)]

....

שימו לב שאם לחייל נגמרו נקודות היעד הוא לא יזוז ולא יתקיף יותר.
ורק כאשר יסתיימו כל הצעדים לכולם – המשחק ייגמר.

קובץ הפלט בסוף המשחק יהיה בפורמט:

Game Configurations

Player1

[נשק (אם יש)], [מיקום], [כמות חיים], [הסוג של חייל 1]

[נשק (אם יש)], [מיקום], [כמות חיים], [הסוג של חייל 2]

....

Player2

[נשק (אם יש)], [מיקום], [כמות חיים], [הסוג של חייל 1]

[נשק (אם יש)], [מיקום], [כמות חיים], [הסוג של חייל 2]

....

Winner is: [השחקן המנצח (מספר) או הכרזה שאין ניצחון]

בהצלחה!