

תרגיל בית מספר 1

נושא: שפת C++ כשפת C משופרת
תאריך הגשה: יום ה', 3/4/2021, 23:59
הגשה ביחידים

בהצלחה רבה!

תיאור התרגיל

מטרתו של תרגיל בית זה הינה כפולה – הראשונה, יישור קו מבחינת רמת התכנות הנדרשת בתחילת הקורס; השנייה, תכנות בשפת C++ כשפת C משופרת, אך לא דווקא כשפה מונחית עצמים בשלב זה. התרגיל יעסוק בבניית מערכת גרפית המסוגלת לצייר קווים ועיגולים, וכן למלא שטחים ולהדפיס את התוצאה הסופית בתווים. המערכת כוללת קלט/פלט, ניהול זיכרון, ואלגוריתמים גרפיים. להנחיות מצורף נספח המספק את האלגוריתמים הנדרשים לפתרון התרגיל.

אפיון

עליכם לתכנן ולממש מערכת גרפית התומכת בפעולות הבאות:

board – הגדרת לוח חדש בגודל row x cols.

plot – ציור נקודה במיקום (x,y) על לוח נתון ובצבע נתון.

line – ציור קו מ-(x1,y1) ל-(x2,y2) על לוח נתון ובצבע נתון.

circle – ציור מעגל עם מרכז (cx,cy) ורדיוס r על לוח נתון ובצבע נתון.

fill – ביצוע פעולת flood fill החל מהנקודה (x,y) על לוח נתון ובצבע נתון.

stamp – שימוש בלוח אחר על מנת "לחתום" על לוח נתון כך ש-(x,y) היא הפינה השמאלית העליונה של החותמת.

print – הדפסת לוח נתון אל ערוץ הפלט הסטנדרטי.

quit – יציאה מהמערכת.

התוכנית קוראת את הפקודות עם הארגומנטים שלהם מערוץ הקלט הסטנדרטי. הפקודות מבוצעות לפי הסדר, עד הפקודה quit או הגעה לסוף הקובץ (end of file). במקרה של פקודה לא חוקית, תודפס הודעת שגיאה לערוץ השגיאה הסטנדרטי, והתוכנית תסיים את פעולתה מיד עם error status. הפקודות נקראות מערוץ הקלט הסטנדרטי תוך התעלמות מרווחים וסופי שורה. אם הפקודה שנקראה אינה מופיעה ברשימה למעלה, יש לצאת עם הודעת שגיאה כאשר XXX הוא הפקודה הלא חוקית שהתקבלה:

unknown command 'XXX'

מפת צבעים

הצבעים במערכת זו הינם לפי הטבלא מוגדרת מראש. כאשר נדרש לתת צבע בפקודה, תמיד הצבע ינתן כמספר צבע לפי הטבלא הבאה. אם מספר הצבע אינו בטבלא, תינתן הודעת שגיאה כאשר <num> הוא מספר הצבע הלא חוקי שניתן:

Invalid color number <num>

מספר הצבע	תו הצבע
0	התו רווח
1	– התו נקודה
2	+ – התו פלוס

תכנות בשפת C++, אביב 2021

3	* – התו כוכבית
4	/ – התו קו נטוי קדמי
5	\ -- התו קו נטוי אחורי
6	– התו קו מאונך
7	- – התו מינוס

להלן אפיון כל אחת מהפקודות הנתמכות.

הגדרת לוח חדש

הפרמטרים של הפקודה הם שם הלוח, ושני מספרים שלמים גדולים מאפס המציינים את מספר השורות והעמודות.

```
board <name> <rows> <cols>
```

לדוגמא:

```
board nice 5 20
```

יגדיר לוח בשם nice עם 5 שורות ו-20 עמודות.

הקואורדינטות על הלוח מתחילות באפס, כלומר x הינו מ-0 עד cols-1, ו-y מ-0 עד rows-1, כאשר (0,0) היא הפינה השמאלית העליונה. גם rows וגם cols צריכים להיות גדולים מאפס, אחת יש להוציא שגיאה:

```
Board: number of rows and columns must be greater that zero
```

התו הראשון בשם הלוח צריך להיות אות (a-z או A-Z) ושאר התווים אותיות או סיפרות. אם שם הלוח לא תקין, יש להוציא את השגיאה הבאה, כאשר XXX שם הלוח שניתן:

```
Invalid board name 'XXX'
```

אם לוח בשם זה כבר קיים, יש לשחרר את הלוח הקודם מהזיכרון ולהגדיר את הלוח החדש.

ציור נקודה

הפרמטרים של הפקודה plot הם שם הלוח, שני מספרים שלמים x, y, ומספר שלם color. הפרמטרים מציינים את שם הלוח עליו יש לצייר את הנקודה, מיקום הנקודה (x,y) ומספר הצבע לפי מפת הצבעים. אם (x,y) מחוץ ללוח, הפקודה לא מבצעת דבר (זו לא נחשבת שגיאה).

```
plot <name> <x> <y> <color>
```

לדוגמא:

```
plot nice 0 0 1
```

יצייר נקודה (צבע מספר 1 במפת הצבעים) בפינה השמאלית העליונה (0,0) של הלוח בשם nice.

אם שם הלוח לא תקין יש להוציא את השגיאה הבאה, כאשר XXX שם הלוח שניתן:

```
Invalid board name 'XXX'
```

אם לא קיים לוח בשם זה, יש להוציא שגיאה:

```
Board named 'XXX' not found
```

אם הצבע אינו בתחום המוגדר במפת הצבעים, יש להוציא שגיאה, כאשר CCC הצבע שניתן בפקודה:

```
Invalid color number CCC
```

תכנות בשפת ++C, אביב 2021

ציור קו

הפרמטרים של הפקודה line הם שם הלוח, 4 מספרים שלמים x_1, y_1, x_2, y_2 , ומספר שלם color. הפרמטרים מציינים את שם הלוח עליו יש לצייר את הקו, את שני הקצוות של הקו (x_1, y_1) ו- (x_2, y_2) , ואת מספר הצבע לפי מפת הצבעים. אין בעייה שאחד הקצוות או שניהם יהיו מחוץ ללוח. הפקודה מציירת רק את החלק שבתוך הלוח.

```
line <name> <x1> <y1> <x2> <y2> <color>
```

לדוגמא:

```
line nice 10 0 0 10 2
```

יצייר על הלוח nice קו בצבע מספר 2 מ- $(10,0)$ ל- $(0,10)$.

ציור הקו יתבצע לפי האלגוריתם של Bresenham המתואר בנספח.

הודעות השגיאה דומות לפקודה plot, כלומר שם לוח לא תקין, לוח לא קיים וצבע לא חוקי.

ציור מעגל

הפרמטרים של הפקודה circle הם שם הלוח, 3 מספרים שלמים cx, cy, r , ומספר שלם color. הפרמטרים מציינים את שם הלוח עליו יש לצייר את המעגל, את מרכז המעגל (cx, cy) , את רדיוס המעגל r ואת מספר הצבע לפי מפת הצבעים. יתכן וכל המעגל או חלקו יהיה מחוץ ללוח, הפקודה מציירת רק את החלק שבתוך הלוח.

```
circle <name> <cx> <cy> <r> <color>
```

לדוגמא:

```
circle nice 10 10 5 2
```

יצייר על הלוח nice מעגל בצבע מספר 2 אשר מרכזו ב- $(10,10)$, ורדיוסו 5.

ציור הקו יתבצע לפי האלגוריתם של Bresenham המתואר בנספח.

הודעות השגיאה דומות לפקודה plot, כלומר שם לוח לא תקין, לוח לא קיים וצבע לא חוקי. בנוסף, אם הרדיוס קטן או שווה לאפס, יש להוציא שגיאה:

Circle: radius must be non-negative

מילוי שטח

הפקודה fill מבצעת מילוי בהצפה (flood fill) החל מנקודה נתונה. החלק המוצף הינו כל שטח הרקע (צבע מספר אפס) אליו ניתן להגיע מנקודת ההתחלה ויחס שכנות 4 בו ארבעת השכנים של הנקודה (x, y) הם $(x \pm 1, y)$ ו- $(x, y \pm 1)$. הנקודות מחוץ ללוח אינן נחשבות כצבע רקע. אלגוריתם אפשרי למימוש מתואר בנספח.

הפרמטרים לפקודה fill הינם שם הלוח עליו יש לבצע את הפעולה, נקודת ההתחלה (x, y) של פעולת המילוי ומספר הצבע לפי מפת הצבעים בו יש להשתמש במילוי.

```
fill <name> <x> <y> <color>
```

לדוגמא:

```
fill nice 5 5 2
```

יבצע מילוי בהצפה על הלוח nice בצבע מספר 2 כאשר נקודת ההתחלה הינה $(5,5)$.

הודעות השגיאה דומות לפקודה plot, כלומר שם לוח לא תקין, לוח לא קיים וצבע לא חוקי. בנוסף, אם מספר הצבע הינו אפס, יש להוציא שגיאה:

FloodFill: cannot flood with the background color

תכנות בשפת C++, אביב 2021

התימה

הפרמטרים של הפקודה stamp הם שם הלוח, שני מספרים שלמים x, y , ושם הלוח נוסף stamp_name. הפרמטרים מציינים את שם הלוח עליו יש להטביע את החותמת, את הנקודה (x, y) המציינת את הפינה השמאלית העליונה של החותמת, ואת שם החותמת. הפעולה מעתיקה את לוח החותמת כפי שהוא כך שהפינה השמאלית העליונה של לוח החותמת תצויר בנקודה (x, y) בלוח המטרה.

```
stamp <name> <x> <y> <stamp_name>
```

לדוגמא:

```
stamp nice 5 5 x
```

תעתיק את הלוח בשם x אל הלוח nice כך שהפינה השמאלית העליונה של x תצויר על nice במיקום (5,5).

הדפסה

לפקודה print יש פרמטר בודד והוא שם הלוח להדפסה.

```
print <name>
```

ההדפסה תבצע באופן הטבעי, כלומר השורות יודפסו החל משורה אפס ($y=0$) ובסוף כל שורה יבוא תו סוף שורה \n. עבור כל שורה התווים יודפסו כפי שהם, כאשר התו עם $x=0$ יודפס ראשון.

יציאה

ישנן שתי דרכים לסיים את התוכנית עבור קלט תקין. הראשונה הינה הגעה לסוף קובץ הקלט, והשנייה הינה הפקודה quit.

דוגמא

הסבר	Output	Input
הגדרת לוח בשם nice עם 5 שורות ו-20 עמודות. ללא פלט.		board nice 5 20
ציור קו על הלוח מהפינה השמאלית העליונה לימנית התחתונה בצבע מספר 3 (כוכבית). ללא פלט.		line nice 0 0 19 4 3
הדפסת הלוח. 5 שורות עם 20 תווים כל אחת. רוב התווים הם רווחים ולכן לא רואים אותם למרות שהם שם.	*** ***** ***** ***** ***	print nice
מילוי השטח בהצפה החל מהפינה השמאלית התחתונה בצבע 2 (הסימן פלוס). זה ממלא את כל השטח מתחת לקו, אבל לא מגיע מעליו. ואז מילוי השטח מעל על ידי צבע 7 (הסימן מינוס).	***----- ++++*****----- +++++++*****----- +++++++*****----- +++++++*****	fill nice 0 4 2 fill nice 19 0 7 print nice
הוספת מעגל עם רדיוס 5 אשר מרכזו בפינה הימנית העליונה. רק החלק אשר בתוך הלוח מצויר.	***-----.- ++++*****-----.- +++++++*****-----.- +++++++*****.- +++++++*****.-	circle nice 19 0 5 1 print nice
יצירת לוח נוסף בשם x בגודל 2x2. ציור קו אלכסוני עליו מהפינה הימנית העליונה לפינה השמאלית התחתונה בצבע '/	/	board x 2 2 line x 0 1 1 0 4 print x
שימוש בלוח הנוסף כחותמת. (2,2) הינו מיקום הפינה השמאלית העליונה של החותמת בלוח nice.	***-----.- ++++*****-----.- ++ /++++*****-----.- ++ /++++*****.- +++++++*****.-	stamp nice 2 2 x print nice

תכנות בשפת C++, אביב 2021

הנחיות עבודה ופורמטים

- על מבנה הנתונים בו מאוחסנים הלוחות והאינפורמציה עליהם להיות מוקצה דינאמית על ה-Heap. יש חופש מוחלט בבחירת טיפוס מבנה הנתונים.
- שם הלוח הינו מחרוזת באורך כלשהו גדול מאפס. השם חייב להתחיל באות אנגלית (גדולה או קטנה), ויתר האותיות צריכות להיות אות אנגלית או סיפרה.
- אפשר להניח שמספר התווים בלוח נכנס ב-int, כלומר קטן מ-31.
- מספר הלוחות שניתן לאחסן בזיכרון, גודלם, ואורך שמם אינו מוגבל בעיקרון (אך בפועל יהיה מוגבל על ידי זיכרון המחשב ועל מגבלות גודל int ושל string).

דגשים

- יש לתכנן מראש את מבנה התכנית, ולהגדיר בהתאם את המרכיבים איתם תעבדו.
- יש לממש את עבודת הקלט והפלט באמצעות רכיבי C++ ולא באמצעות פונקציות C.
- הקצאה דינאמית צריכה להתבצע רק באמצעות הפקודות new ו-delete.
- עבודה עם מחרוזות יכולה להתבצע באמצעות טיפוס `std::string`, אך אין דרישה לכך.
- **בתרגיל בית זה אין להשתמש בספריית STL; מבני הנתונים והאלגוריתמים צריכים להיות ממומשים על-ידיכם.**
- יש לבדוק תקינות קלטים ולהציג הודעות שגיאה מתאימות.
- עליכם לוודא כי התכנית עוברת קומפילציית g++ התואמת את הקומפיילר שעל שרת החוג ללא כל שגיאות או אזהרות כלשהן, ורצה בהצלחה.
- עליכם לתעד את הקוד באמצעות הערות המתארות בקצרה את המודולים והפונקציות השונים.
- יש להריץ את הבדוק האוטומטי על שרת החוג בטרם ההגשה בכדי לוודא תאימות ונכונות של ההגשה: **hwcheck** (או לחילופין עשו שימוש בפרוטוקול HTML בקישור <https://cs.telhai.ac.il/homework>).

הגשה

- עליכם להגיש במערכת Moodle קובץ ארכיב מטיפוס zip בלבד, ששמו כולל את קוד הקורס (32), שם התרגיל (ex1) ותעודת הזהות של הסטודנט/ית המגיש/ה, מופרדים בקו תחתי בפורמט הבא: **32_ex1_studID.zip**.
- על ארכיב zip זה להכיל את כל קבצי המקור (ממשק/מימוש) הנדרשים לקומפילציה, והוא רשאי להכיל תיעוד טקסטואלי; מבחינת טיפוס קבצים, עליו לכלול רק קבצים עם סיומות ***.txt *.h *.cpp**.
- לדוגמה: על סטודנט שמספר הזיהוי שלו הינו 012345678 להגיש ארכיב בשם **32_ex1_012345678.zip** הכולל את כל קבצי המקור של הפרוייקט, ללא תיקיות כלשהן, ורשאי להכיל קובץ טקסטואלי לתיעוד.

אי-הקפדה על ההנחיות, כולל פורמט ההגשה הדיגיטלי, תגרור הורדה בציון התרגיל.
לא תתקבלנה הגשות באיחור!

בנספח זה ניתנים האלגוריתמים הנדרשים למימוש המערכת.

האלגוריתם של Bresenham לציור קו

האלגוריתם הינו לציור קו מהנקודה (x_1, y_1) לנקודה (x_2, y_2) .

```
input: x1, y1, x2, y2

plotyx = false;

if |x2-x1| < |y2-y1| {
    swap x1 y1
    swap x2 y2
    plotyx = true
}

if x2 < x1 {
    swap x1 x2
    swap y1 y2
}

dx = x2 - x1
dy = |y2 - y1|
sy = sign(y2 - y1) // sign(z) = 1 if z>0; -1 if z<0; 0 otherwise

e = -dx
x = x1, y = y1
while (x ≤ x2) {
    if plotyx then plot(y,x) else plot(x,y)
    x += 1
    e += 2 * dy
    if (e > 0) {
        y += sy
        e -= 2 * dx
    }
}
```

האלגוריתם של Bresenham לציור מעגל

האלגוריתם הינו לציור מעגל ברדיוס r עם מרכז (cx, cy) .

הפונקצייה $plot8(cx, cy, x, y)$ מציירת את 8 הנקודות $(cx \pm x, cy \pm y)$ ו- $(cx \pm y, cy \pm x)$.

תכנות בשפת C++, אביב 2021

```
input: cx, cy, r

x = 0, y = r
d = 3 - 2 * r
plot8(cx, cy, x, y)
while (y >= x) {
    x += 1
    if (d > 0) {
        y -= 1
        d += 4 * (x - y) + 10
    } else {
        d += 4 * x + 6
    }
    plot8(cx, cy, x, y)
}
```

אלגוריתם למילוי שטחים – Flood fill

הערה: ישנם אלגוריתמים יעילים יותר הזה המוצג, אבל למערכת הגרפית המתוארת, אין בהם צורך.

אלגוריתם רקורסיבי לביצוע flood fill המשתמש בשכנות 4 החל מהנקודה (x,y) . מקור [wikipedia flood fill](https://en.wikipedia.org/wiki/Flood_fill).

```
function flood(x,y)
    if (x,y) is not background or outside the frame
        return
    plot(x,y)
    flood(x+1,y)
    flood(x-1,y)
    flood(x,y+1)
    flood(x,y-1)
```