

פרויקט 1

מטרה

מימוש המשחק BlackJack הידוע בכינויו 21, באמצעות OOP.

חוקי המשחק

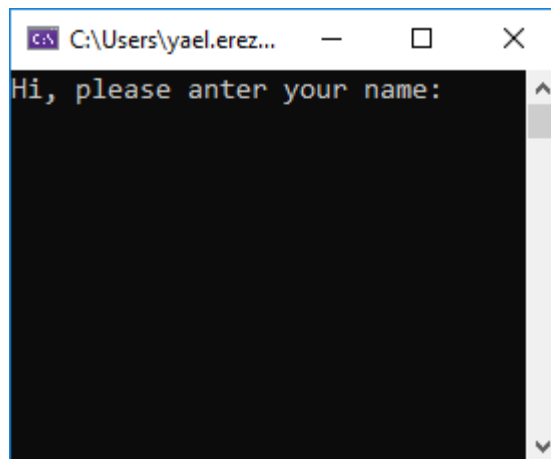
במשחק יש 52 קלפים:

- קלפי מספר מ-2 עד 10. 4 מכל אחד. ערכם הוא כערך המספרי הנקוב בהם.
- 4 נסיכים (ערך מספרי 11), 4 מלכות (ערך מספרי 12) ו-4 מלכים (ערך מספרי 13), נקראים פרצוף. ערכם במשחק הוא 10 ולא הערך הנקוב.
- 4 אסים. ערכם 1 או 11 לפי הצורך.

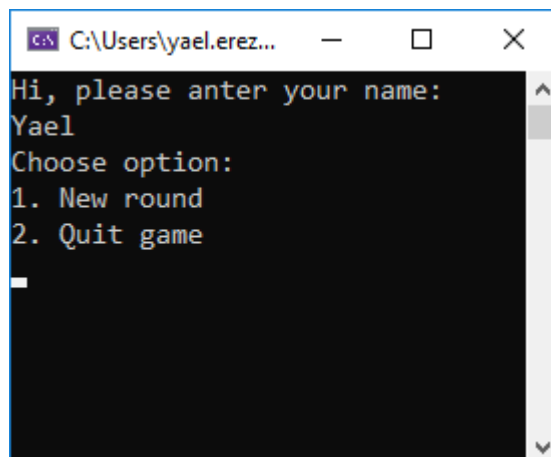
שחקן משחק מול הדילר, כאשר המטרה היא לצבור קלפים שערכם הכולל כמה שיותר קרוב ל-21. מעל 21 השחקן מפסיד.

מהלך המשחק:

1. שואלים את השחקן לשמו



2. שואלים את השחקן אם הוא רוצה סיבוב נוסף או לצאת מהמשחק?



אם השחקן בחר 1, הקלפים מהסיבוב הקודם מוחזרים לחפיסה ומערבבים אותה. הדילר לוקח לעצמו 2 קלפים ומחלק לשחקן 2 קלפים.

```

C:\Users\yael.erez...
Hi, please enter your name:
Yael
Choose option:
1. New round
2. Quit game
1
Dealer cards:
9 7 (Total of 16)
Yael cards:
Q 9 (Total of 19)

```

אם השחקן בחר 2 המשחק מסתיים.

```

Microsoft Visual Studio Debug Console
Hi, please enter your name:
Yael
Choose option:
1. New round
2. Quit game
2
Out of 0 rounds, you won 0 times and lost 0 times
Bye Bye

```

3. הדילר והשחקן משחקים לסירוגין, כאשר הדילר ראשון. כל אחד בתורו מחליט אם הוא לוקח קלף אחד נוסף. הדילר לוקח קלף נוסף אם סך הקלפים שלו לא עובר את הערך 16. השחקן יבחר אם לקחת קלף נוסף או לא. אחרי כל לקיחת קלף, אם עברו את הערך 21 אותו שחקן מפסיד, והשני מנצח. אם גם הדילר וגם השחקן בוחרים לא לקחת קלף נוסף, מנצח השחקן שסכום קלפיו קרוב יותר ל 21. להלן כמה הרצות אפשריות:

```

Microsoft Visual Studio Debug Console
Hi, please enter your name:
Yael
Choose option:
1. New round
2. Quit game
1
Dealer cards:
9 J (Total of 19)
Yael cards:
4 5 (Total of 9)
Choose option:
1. Take a card from pile
2. No more cards
1
Yael cards:
4 5 10 (Total of 19)
Choose option:
1. Take a card from pile
2. No more cards
2
It's a tie! No one wins
Choose option:
1. New round
2. Quit game
2
Bye Bye

```

```
C:\Windows\system32\cmd.exe
Hi, please enter your name:
Yael
Choose option:
1. New round
2. Quit game
1
Dealer cards:
J J (Total of 20)
Yael cards:
7 7 (Total of 14)
Choose option:
1. Take a card from pile
2. No more cards
2
Dealer wins
Choose option:
1. New round
2. Quit game
2
Bye Bye
```

```

C:\Windows\system32\cmd.exe
Hi, please enter your name:
Yael
Choose option:
1. New round
2. Quit game
1
Dealer cards:
10 6 (Total of 16)
Yael cards:
10 10 (Total of 20)
Dealer cards:
10 6 10 (Total of 26)
Yael wins
Choose option:
1. New round
2. Quit game
2
Bye Bye

```

לאחר כל סיבוב חוזרים למספר 2.

לרשותכם קובץ BlackJack.exe לבצע כמה הרצות שתמצאו...

מחלקות

התוכנית תכיל את המחלקות הבאות:

קלף - Card

קלף מזוהה ע"י הערך שלו – מספר בין 1 ל 13. (מומלץ להגדיר ערכים אלו כקבועים הניתנים לשינוי בקלות).
הערך של הקלף מועבר למחלקה ביצירתה.
מחלקה זו תהיה בסיס לקלפים בפרויקט 2, הקפידו על המימוש כפי שמתואר להלן:

למחלקה יהיו את המאפיינים הסטטיים הבאים:

- סך כל הקלפים שנוצרו
- מערך המתאר כמה קלפים נוצרו מכל סוג

כמו כן למחלקה יהיה מאפיינים שהם `const static`:

- המספר של הקלפים שיש ליצור (במקרה שלנו 52)
- המספר שיש ליצור מכל סוג (במקרה שלנו 4)

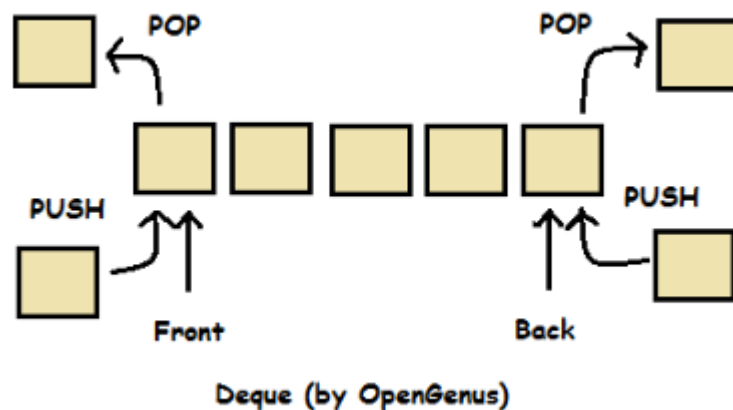
המחלקה תממש את הפעולות הבאות:

- פונקציה בונה
- פונקציה הורסת
- Copy Constructor
- מתודה סטטית אשר מגרילה ערך לקלף: `static int toss_val();`
- המתודה תדאג להגריל ערך חוקי לפי הערכים של המשתנים הסטטיים.
- המחלקה תדע להדפיס את עצמה ע"י מימוש `operator<<`.

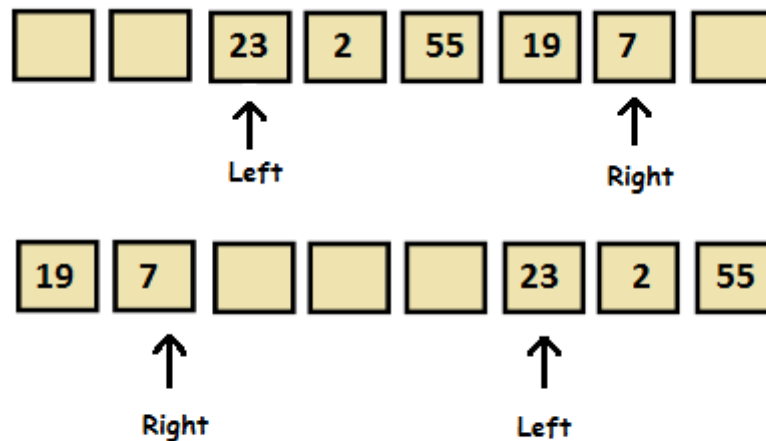
ניתן להוסיף פונקציות נוספות ע"פ הצורך.

תור דו צדדי – Dequeue

תור דו צדדי הוא תור שניתן לבצע עליו פעולות `push` ו `pop` משני הצדדים. הוא ישמש אותנו כחבילת קלפים, ולכן הוא יהיה תור של מצביעים לקלפים. (הקפידו לאחסן בו מצביעים לקלפים ולא קלפים כדי שתוכלו להשתמש בו גם בפרויקט 2).



התור ימומש כמערך ציקלי:



Deque (by OpenGenus)

למחלקה Dequeue יהיו המאפיינים הבא:

m_size	מספר האיברים בתור
m_capacity	גודל המערך
m_queue	מערך של מצביעים לקלפים באורך m_capacity
m_left	האינדקס של האיבר הראשון בתור

המחלקה תממש את הפעולות הבאות:

- פונקציה בונה
- פונקציה הורסת
- הכנסת איבר לסוף התור (push_back)
- הוצאת איבר מסוף התור (pop_back)
- הכנסת איבר לראש התור (push_front)
- הוצאת איבר מראש התור (pop_front)
- החזרת האיבר שבראש התור (front)
- החזרת האיבר שבסוף התור (back)
- החזרת מספר האיברים בתור (size)
- חיווי לתור ריק (is_empty)

ניתן להוסיף פונקציות נוספות ע"פ הצורך.

שחקן – Player

מחלקה המתארת שחקן, אנו נשתמש בה גם בפרויקט 2. למחלקה זו יהיו השדות הבאים:

1. שם השחקן
2. הערך של סך הקלפים שבידי השחקן
3. מספר הסיבובים בהם השחקן ניצח
4. מספר הסיבובים בהם השחקן הפסיד

המחלקה תדע להדפיס את עצמה ע"י מימוש <<operator.

BlackJack – ג'ק – בלאק-ג'ק

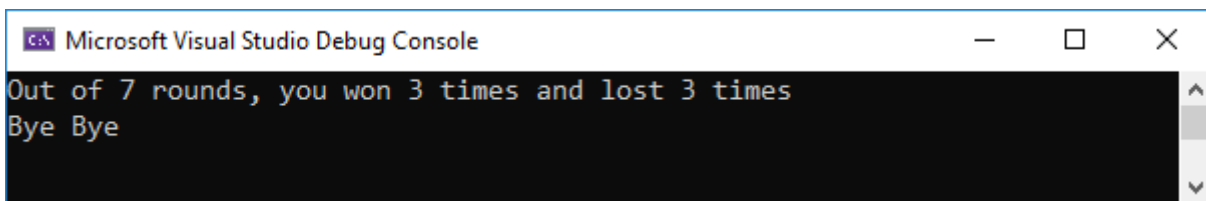
המחלקה שמממשת את המשחק. אנו נממש משחק של שחקן אחד מול המחשב (הדילר).

למחלקה יהיו המאפיינים הבאים:

1. מצביע לשחקן
2. ערימת הקלפים שלא חולקו (Dequeue)
3. הקלפים של השחקן
4. הקלפים של הדילר
5. מספר הסיבובים במשחק

בסוף המשחק המחלקה תדפיס כמה משחקים השחקן ניצח וכמה הפסיד.

פלט התוכנית יכול למשל להראות כך:



```
Microsoft Visual Studio Debug Console
Out of 7 rounds, you won 3 times and lost 3 times
Bye Bye
```

(במשחק זה היה סיבוב אחד שהסתיים בתיקו).

עזרה בהגדרת מספר: בצעו עזרה מקוונת על `rand` ו-`srand`
<http://www.cplusplus.com/reference/cstdlib/rand>

בסמסטר זה העבודות יבדקו גם באופן ידני וגם באופן אוטומטי.

הבדיקה האוטומטית:

הבודק האוטומטי מקמפל, מריץ ובודק דליפות זכרון. אנא הקפידו על ביצוע מדויק של ההגשה, כדי שהבדיקה האוטומטית תעבוד (שמות קבצים וכו). על הקוד שלכם לעבוד עם ה `main` הבא:

```
#include "BlackJack.h"
#include <time.h>

int main()
{
    srand(time(NULL));
    BlackJack bj;

    bj.play();
    return 0;
}
```


שימו לב לא להשתמש בקטעי קוד שעובדים רק ב windows או ב dos כמו:

```
scanf_s
#include <conio.h>
getch
kbhit
#pragma once
```

הבדיקה הידנית תתמקד בסגנון:

- לכל מחלקה קבצי h. ו- cpp. נפרדים
- על התוכנית להיות כתובה באופן מסודר ומדורג (הזזה ימינה עם פתיחה של כל סוגריים מסולסלים)
- יש לתת שמות משתנים ופונקציות משמעותיים
- יש להוסיף הערות הסבר באנגלית בתוכנית (נא לא להגזים, קצר וקולע)
- המנעו משכפול קוד ומפונקציות ארוכות
- ערכים קבועים הגדירו כמשתנה const-י
- במקומות שצריך הקפידו להגדיר מתודות כ const ולהעביר אובייקטים by reference
- שימוש ברשימות אתחול

אופן ההגשה :

1. הגשה בזוגות.
 2. העבודה תוגש **גם** במודל **וגם** מודפסת ומשודכת באופן נוח לקריאה.
 3. יש להגיש מסמך וורד כמפורט בהמשך ואת כל קבצי הקוד שכתבתם מלבד ה main
 4. על הקבצים להיות מכווצים בתוכנת zip ולא בשום תוכנת כיווץ אחרת.
 5. שם קובץ ה zip יהיה ה ת.ז. של הסטודנטים כך: id1_id2.zip.
 6. הסבר על מסמך הוורד:
- a. בעמוד הראשון כותרת העבודה בפורמט הבא:

<p>עבודה ב OOP 31695 סמסטר א' תש"פ, עבודה 1</p> <p>תאריך ההגשה: dd/mm/yyyy</p> <p>שם הסטודנט, תעודת זהות, מייל: שם משפחה, שם פרטי, Email</p> <p>שם הסטודנט, תעודת זהות, מייל: שם משפחה, שם פרטי, Email</p> <p>ציון:</p> <p>-----</p>
--

- b. בהמשך העמוד הראשון:
- תחת הכותרת "תאור התוכנית" תאור מילולי טכני ותמציתי של מחלקות התוכנית (Function members ו Data members) והשיטה/האלגוריתם שבה נכתבה התוכנית.
- התאור צריך להיות כזה שאם הוא ינתן למתכנת אחר ביחד עם הגדרת הבעיה המתכנת יכתוב תוכנית שפועלת באותה השיטה כמו התוכנית שלכם.
- c. תחת הכותרת "מקורות" רשימה של כל המקורות בהם נעזרתם בכתיבת התוכנית
- d. תחת הכותרת "בעיות ידועות" רשימה של הבעיות הידועות בתוכנית (אם קיימות)

צוות הקורס שומר לעצמו את הזכות לזמן את הסטודנטים לבחינה בעל פה על העבודה שהוגשה.