

סמסטר ב' תשע"ד  
מועד: א' 11/06/2014  
שעה: 9:00  
חלק א': שעה ועשרים דקות  
חומר עזר: כל חומר עזר כתוב מותר

מספר זהות:

--	--	--	--	--	--	--	--	--	--

**בחינה בקורס: תכנות מכוון עצמים ושפת C++**

**מרצים: אמיר קירש, ד"ר אלון שקלר**

**מדבקת  
ברקוד**

**חנחיות כלליות לבחינה:**

- המבחן מורכב משני חלקים:
- חלק א' כולל 6 שאלות אמריקאיות. משקל כל שאלה 8 נקודות, סה"כ: 48 נק'.
- חלק ב' כולל שאלת תכנות שמשקלה הכולל 52 נק'.
- לכל חלק יינתן זמן בנפרד. חלק א' הינו עם חומר פתוח, חלק ב' הינו עם חומר סגור.

**חנחיות לחלק א':**

- בשאלות האמריקאיות יש לסמן תשובה אחת לכל שאלה בטבלה המצורפת. במידה ומספר תשובות נראות נכונות יש לסמן את התשובה הנכונה ביותר.
- בסיום המבחן יש לרשום מס' ת.ז. על גבי טופס המבחן, לוודא שטבלת התשובות האמריקאיות נמצאת יחד עם טופס המבחן ולחגישם. אין להגיש מחברות או טיוטות בחלק א' אלא רק את טופס המבחן.
- המבחן הינו עם חומר פתוח. כל חומר עזר כתוב מותר למעט מכשירים אלקטרוניים למיניהם. אין להעביר חומר עזר בין תלמידים במהלך המבחן.

**בהצלחה !**

טבלת תשובות לחלק האמריקאי – טור A

א	ב	ג	ד	ה	ו

שאלה 1
שאלה 2
שאלה 3
שאלה 4
שאלה 5
שאלה 6

**חובה** לספק חסבר במידה ונבחרה תשובה ו' (אף תשובה אינה נכונה).

מותר לצרף חסבר גם עבור תשובות אחרות. אמנם רק תשובה נכונה תזכה בניקוד עבור כל שאלה, אולם ניתן יהיה להסתמך על החסבר במסגרת ערעור, אם יידרש. מומלץ לצרף חסבר לתשובה אמריקאית במיוחד במקרים בהם נראה לך שתשובתך דורשת חסבר או נימוק.

שאלה 1 :

---

שאלה 2 :

---

שאלה 3 :

---

שאלה 4 :

---

---

---

שאלה 5 :

---

---

---

שאלה 6 :

---

---

---

חלק א' – שאלות אמריקאיות (48 נק' – 8 נק' לכל שאלה)

שאלות 1-6 מתייחסות לקטע הקוד הבא:

```

1.  template<class E> class Array
2.  {
3.      class Element
4.      {
5.          Array<E>* const pArray; // pointer can't move
6.          const int index;
7.      public:
8.          Element(Array<E>* p, int i)
9.              : pArray(p), index(i) {}
10.         const Element& operator=(const E& e) {
11.             pArray->set(index, e); // call copy-on-write
12.             return *this;
13.         }
14.         operator E() const {
15.             return pArray->arr[index];
16.         }
17.     };
18.
19.     friend class Element;
20.     E* arr;
21.     int size;
22.     int* ref_counter;
23.     void attach(const Array& a) {
24.         arr = a.arr; size = a.size;
25.         ref_counter = a.ref_counter;
26.         ++(*ref_counter);
27.     }
28.     void detach() {
29.         if(--(*ref_counter) == 0) {
30.             delete []arr;
31.             delete ref_counter;
32.         }
33.     }
34.     void set(int index, const E& e) {
35.         if(*ref_counter > 1) { // need copy-on-write!
36.             Array temp = clone();
37.             detach();
38.             attach(temp);
39.         }
40.         arr[index] = e;
41.     }
42. public:
43.     explicit Array(int);

```

```

44.     Array<E> clone() const;
45.     Array(const Array<E>& a){attach(a); }
46.     ~Array(){detach(); }
47.     const Array& operator=(const Array<E>& a) {
48.         detach(); attach(a); return *this;
49.     }
50.     Element operator[](int index) {
51.         return Element(this, index);
52.     }
53.     const E& operator[](int index) const {
54.         return arr[index];
55.     }
56. };
57.
58. template<class E>
59. Array<E>::Array(int size1)
60.     : size(size1), ref_counter(new int(1))
61. {
62.     arr = new E[size];
63. }
64.
65. template<class E>
66. Array<E> Array<E>::clone() const {
67.     Array temp(size);
68.     for(int i=0; i<size; ++i) {
69.         temp.arr[i] = arr[i];
70.     }
71.     return temp;
72. }
73.
74. int main()
75. {
76.     Array<int> intArr1(5), intArr2(10);
77.     Array<char*> strArr1(3), strArr2(3);
78.     intArr1[0] = intArr2[0] = 1;
79.     strArr1[0] = strArr2[0] = "Hello";
80.     intArr2[1] = 2;
81.     strArr1 = strArr2;
82.     strArr2[1] = " world!";
83.     intArr2 = intArr1;
84.     cout << intArr2[0] << ", " << intArr2[1] << endl;
85.     cout << strArr2[0] << strArr2[1] << endl;
86.     return 1;
87. }

```

נתון שהתוכנית לעיל עוברת קומפילציה בהצלחה!

### שאלה 1

כמה פעמים התוכנית עוברת בפונקציה clone שבשורה 65?

- א. אף פעם.
- ב. פעם אחת.
- ג. פעמיים.
- ד. שלוש פעמים.
- ה. ארבע פעמים!
- ו. אף תשובה אינה נכונה.

### שאלה 2

כמה פעמים התוכנית נכנסת לפונקציה set שבשורה 34?

- א. שלוש פעמים.
- ב. ארבע פעמים.
- ג. שש פעמים.
- ד. שמונה פעמים.
- ה. עשר פעמים.
- ו. אף תשובה אינה נכונה.

### שאלה 3

כמה פעמים התוכנית עוברת בשורה 30?

- א. פעמיים.
- ב. שלוש פעמים.
- ג. ארבע פעמים.
- ד. חמש פעמים.
- ה. זה לא ידוע, מכיון שאנחנו לא יודעים כיצד מבוצעות החשמות של שורות 78 ו-79.
- ו. אף תשובה אינה נכונה.

#### שאלה 4

מה יודפס בשורה 84?

- א. 1, 1
- ב. 1, 2
- ג. 1, <some garbage number>
- ד. <some garbage number>, 2
- ה. <some garbage number>, <some garbage number>
- ו. אף תשובה אינה נכונה.

#### שאלה 5

מה יודפס בשורה 85?

- א. Hello world!
- ב. Hello
- ג. world!
- ד. Hello Hello
- ה. Hello!
- ו. אף תשובה אינה נכונה.

שאלה 6

מורן, שירן, לירן וטירן חתכוננו למבחן ועברו על הקוד לעיל.

לירן טען : ללא החכרזה על friend בשורה 19, שורה 15 לא היתה עוברת קומפילציה.

שירן טען : ללא החכרזה על friend בשורה 19, שורה 11 לא היתה עוברת קומפילציה.

טירן טען : ב-main שבמבחן אנחנו לעולם לא עוברים באופרטור [] ה-const-י (של שורה 53) כיון שבמחלק התוכנית אין אובייקט Array שמוכר כ-const ומנסה להפעיל אופרטור [].

מורן טען : לא ניתן בתוכנית זו לייצר מערך באמצעות התחביר הבא :

```
Array<int> a1 = 1;
```

מי מהחבורה צודק ?

- א. לירן וטירן.
- ב. טירן ומורן.
- ג. לירן ושירן.
- ד. לירן, שירן ומורן.
- ה. כולם צודקים.
- ו. אף אחת מהתשובות אינה נכונה.

סמסטר ב' תשע"ד  
מועד: א' 11/06/2014  
שעה: 11:00  
חלק ב': 2½ שעות  
חומר עזר: כל חומר עזר אסור

מספר זהות:

--	--	--	--	--	--	--	--	--	--

**בחינה בקורס: תכנות מכוון עצמים ושפת C++**

**מרצים: אמיר קירש, ד"ר אלוו שקלר**

**מדבקות  
ברקוד**

**הנחיות כלליות לבחינה:**

- המבחן מורכב משני חלקים:  
חלק א' כולל 6 שאלות אמריקאיות. משקל כל שאלה 8 נקודות, סה"כ: 48 נק'.  
חלק ב' כולל שאלת תכנות שמשקלה ככולל 62 נק'.  
לכל חלק יינתן זמן בנפרד. חלק א' הינו עם חומר פתוח, חלק ב' הינו עם חומר סגור.

**הנחיות לחלק ב':**

- חלק ב' הינו עם חומר סגור.
- חובה לתעד בשאלת התכנות כל פעולה לא ברורה שנעשית.
- נא לכתוב בכתב קריא ולא מחובר.
- יש לענות באופן מסודר לפי סעיפי השאלה.

**בהצלחה !**



**חלק ב' – שאלת תכנות (52 נק')**

עליך לממש את מערך הטיסות והחזמנות של חברת התעופה האוסטרלית "קנגה" ללא ממשק משתמש. החברה מחזיקה בצי מטוסים ומפעילה טיסות באוסטרליה וליעדים שונים בעולם. לחברה צי מטוסים אליו מתווספים מטוסים חדשים או נגרעים מטוסים.

כל מטוס בצי חינו מדגם מסוים. דגם מאופיין ע"י: שם הדגם, אורך המטוס, אורך מוטת הכנפיים ואורך מסלול נחיתה נדרש. בנוסף, לכל דגם ישנו מספר מושבים המסודרים ב-L שורות של K מושבים. במושבים מסוימים ישנם תנאים מיוחדים כגון: מעבר חירום – איסור חושבת ילדים מתחת לגיל 16, מושב לפני מעבר חירום – הגבלה בחשכבת המושב, מושב עם מרווח רגליים גדול – מתאים לכרטיסים בתעריף C ומעלה בלבד (התעריפים יפורטו בהמשך). מושב יכול לקיים מספר תנאים מיוחדים (למשל: מושב שחוא לפני מעבר חירום ועם מרווח רגליים גדול וכד').

לוח הטיסות של החברה נקבע מראש באופן שבו לתאריך מסוים ישנה טיסה מיעד א' ליעד ב' במטוס מדגם X, וכן משויך לטיסה המטוס הספציפי מדגם זה, אך במידת הצורך ניתן להחליף את השיבוץ למטוס אחר מאותו הדגם לפני הטיסה עצמה, אך לא למטוס מדגם אחר.

מחיר מושב יכול להיות לפי אחת מ-5 קטגוריות מחיר, המסומנות A עד E כאשר A הוא התעריף הזול ביותר ו-E חיקר ביותר. בכל טיסה החברה מקצה מספר מקומות מכל תעריף, אבל בלי לשייכם מראש למקומות בטיסה.

הזמנת מקומות בטיסה מתבצעת תמיד כחזמנה של טיסה בכיוון אחד ומורכבת מ:

- בחירת מקור ויעד (למשל: מלבורן לסידני)
- טווח תאריכים אפשרי לחיפוש (ניתן לחפש רק עד שנה אחת קדימה)
- פרטים של לקוח אחד (או יותר במקרה שמדובר בקבוצה): שמות ותאריכי לידה (חובה), מספר לקוח במועדון הנוסע המתמיד (אם יש)

על המערכת להחזיר את הטיסות המתאימות לפי נתוני החיפוש. נתוני הטיסות צריכים לכלול את הנתונים הבאים: פוינטר לאובייקט הטיסה הרלבנטי וכן מחיר לכרטיסים בקטגורית המחיר הזולה ביותר האפשרית בטיסה כפי שמתואר בהמשך.

המחירים שיחושבו עבור כל טיסה יחיו לפי התעריף הזול ביותר, לדוגמא: אם כל המושבים מתעריפים A, B נמכרו, אזי התעריף שיוצג הוא C. כמו-כן, אם חלקו מבקש מספר מקומות, על כולם להיות באותה קטגוריית מחיר כלומר גם אם נשארו כרטיסים בקטגוריה זולה יותר אך לא בכמות מספקת, יוקצו מקומות מקטגוריה יקרה יותר במחלקה זו. אם אין מספיק מקומות באף תעריף, הטיסה לא תצורף לתוצאות החיפוש.

מחיר כרטיס בקטגוריה מסוימת יחושב באמצעות פונקציה סטטית במחלקה קיימת (שאינך נדרש לממש או להציג):

```
double PriceCalculator::calcTicketPrice(Flight* f, const string& priceCategory);
```

כאשר המשתמש יבחר טיסה מסוימת המערכת תאפשר לו לבחור מקומות בטיסה ע"י פירוט המושבים הפנויים בקטגוריה הזולה האפשרית. לצורך כך יש לייצר פונקציות מתאימות, עם החתימות הללו:

```
list<SeatData> Flight::getAvailableSeats(Flight* f, list<CustomerData>, const string& priceCategory) const;
```

```
ReservationResult Flight::reserveSeats(Flight* f, list<SeatReservation> seats);
```

חלק מהמחלקה SeatData יכיל את התנאים המיוחדים של המושב אם יש כאלה - עליכם להחליט כיצד לייצגם. המחלקה SeatReservation תכיל בין היתר את נתוני חלקו עבור המושב שנבחר.

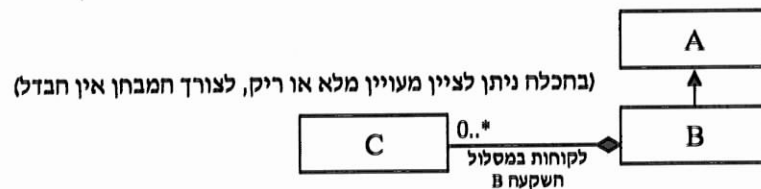
מבחינת עבודה עם תאריכים במידת הצורך ניתן להגדיר מחלקת תאריך עם data members מתאימים ו- prototypes של מתודות (ולחציגה בסעיף ב') – וניתן להשתמש בה בלי צורך לממש.

שימו לב:

במבחן זה אינכם נדרשים לכתוב ממשק משתמש, לכן כל פניה למשתמש לקבלת קלט וכל חדפסה למסך ייחשבו כשגיאת ויפחיתו נקודות.

סעיף א' (10 נקודות)

הצג דיאגרמת מחלקות של כל המחלקות הנדרשות לפתרון.  
על דיאגרמת המחלקות להציג קשרים של חורשה וחכלה בין מחלקות.  
בדוגמא למטה, למשל, B נורש מ-A ו-B מכיל בין 0 לאינסוף איברים מסוג C.



עבור קשר הכלה יש לציין על גבי חקו או מתחתיו את סיבת/מחות החכלה.  
בסעיף זה, החתייחסות להכלה היא זחה בין אם מדובר בחצבעה או בחכלה ממש.

סעיף ב' (12 נקודות)

כתוב את הגדרת כל המחלקות הנדרשות לפתרון (כל ה-prototypes כולל כאלו הנדרשים עבור סעיפי המימוש שלחלן, ללא המימושים עצמם).  
הקפד על שימוש נכון ב-const, private, protected, public.

סעיף ג' (16 נקודות)

ממש באופן מלא את הפונקציות הנדרשות לחיפוש טיסות (6 נק'), הפונקציה getAvailableSeats (5 נק') והפונקציה reserveSeats (5 נק'). יש לחשתמש לפי הצורך בפוני עזר של סעיף ח' לחלן.

סעיף ד' (9 נקודות)

ממש את הפעולות של הוספת מטוס וגריעת מטוס ממערך המטוסים (5 נק'), אין לאפשר גריעת של מטוס שמשויך לטיסה עתידית, ניתן לחשתמש ב-Exception. וכן את מערך הטיסות (4 נק') כולל הוספת טיסה וחקצאת מטוס לטיסה (חניחו כי ישנה פונקציה שעליכם לקרוא לה אך אינכם צריכים לממש אשר בודקת אם המטוס פנוי במועד הטיסה).

סעיף ה' (5 נקודות)

ממש את הטיפול הכללי בתנאים מיוחדים עבור מושבים בטיסה, כולל מימוש ספציפי אחד בלבד, של התנאי עבור מעבר חירום: כאשר ישנו ילד מתחת לגיל 16 בקבוצה, אסור לחציע מקומות בשורה של יציאת חירום.