מדינת ישראל
משרד החינוך

סוג הבחינה: בגרות
מועד הבחינה: קיץ תשע"ז, 2017
מספר השאלון: **899222, 602**
תרגום לערבית (2)

دولة إسرائيل
وزارة التّربية والتّعليم

نوع الامتحان: بچروت
موعد الامتحان: صيف 2017
رقم النّموذج: **899222، 602**
ترجمة إلى العربيّة (2)

מדעי המחשב

2 יחידות לימוד

علم الحاسوب

وحدتان تعليميّتان

הוראות לנבחן

א. משך הבחינה: שלוש שעות.

ב. מבנה השאלון ומפתח ההערכה:
בשאלון זה שלושה פרקים.
פרק ראשון: יש לענות על חמש השאלות 1-5,
לכל שאלה — 10 נק' (10x5) — 50 נק'
פרק שני: יש לענות על שתיים מהשאלות 6-8,
לכל שאלה — 15 נק' (15x2) — 30 נק'
פרק שלישי: יש לענות על אחת מהשאלות 9-10,
לשאלה — 20 נק' (20x1) — 20 נק'
סה"כ — 100 נק'

ג. חומר עזר מותר בשימוש:
כל חומר עזר, חוץ ממחשב הניתן לתכנות.

ד. הוראות מיוחדות:
(1) כתוב בשפה אחת בלבד את כל התכניות שאתה נדרש לכתוב.
(2) **רשום על הכריכה החיצונית** של המחברת את השפה שבה אתה כותב — Java **או** C#.
הערה: בתכניות שאתה כותב לא יורדו לך נקודות, אם תכתוב אות גדולה במקום אות קטנה או להפך.

تعليمات للممتحَن

أ. مدّة الامتحان: ثلاث ساعات.

ب. مبنى النّموذج وتوزيع الدّرجات:
في هذا النموذج ثلاثة فصول.
الفصل الأوّل: يجب الإجابة عن خمسة الأسئلة 1-5،
لكلّ سؤال – 10 درجات (10x5) – 50 درجة
الفصل الثاني: يجب الإجابة عن اثنين من الأسئلة 6-8،
لكلّ سؤال – 15 درجة (15x2) – 30 درجة
الفصل الثالث: يجب الإجابة عن أحد السؤالين 9-10،
للسؤال – 20 درجة (20x1) – 20 درجة
المجموع – 100 درجة

ج. موادّ مساعدة يُسمح استعمالها: أيّة مادّة مساعِدة، عدا الحاسوب الذي يمكن برمجته.

د. تعليمات خاصّة:
(1) اكتب جميع البرامج التي يُطلب منك كتابتها، بلغة واحدة فقط.
(2) **اكتب على الغلاف الخارجيّ** لدفتر الامتحان اللغة التي تكتب بها – Java أو C#.
ملاحظة: في البرامج التي تكتبها لن تُخصم درجات إذا كتبت حرفًا كبيرًا مكان حرف صغير أو بالعكس.

اكتب في دفتر الامتحان فقط، في صفحات خاصّة، كلّ ما تريد كتابته مسوّدة (رؤوس أقلام، عمليّات حسابيّة، وما شابه).
اكتب كلمة "مسوّدة" في بداية كلّ صفحة تستعملها مسوّدة. كتابة أيّة مسوّدة على أوراق خارج دفتر الامتحان قد تسبّب إلغاء الامتحان!

التّعليمات في هذا النّموذج مكتوبة بصيغة المذكّر وموجّهة للممتحَنات وللممتحَنين على حدّ سواء.

ב ה צ ל ח ה ! **نتمنّى لك النّجاح!**

الأسئلة

انتبه: عليك كتابة جميع البرامج التي يُطلب منك كتابتها، بلغة واحدة فقط.
اكتب على الغلاف الخارجيّ لدفتر الامتحان اللغة التي تكتب بها – Java أو C#.
ملاحظة: في كلّ سؤال يُطلب فيه استقبال، لا حاجة لفحص سلامة المدخَلات.
للّذين يحلّون بلغة Java: في كلّ سؤال يُطلب فيه استقبال، افترِضْ أنّ الأمر التالي مكتوب في البرنامج: Scanner input = new Scanner (System.in);

الفصل الأوّل (50 درجة)

أجب عن خمسة الأسئلة 1-5 (لكلّ سؤال – 10 درجات).

1. أمامك عمليّة مكتوبة بلغة Java وبلغة C#.

Java

```
public static int  mystery ( int x, int y, int z)
{
    int temp = x;
    if (y > temp) { temp = y; }
    if (z > temp) { temp = z; }
    return temp;
}
```

C#

```
public static int  Mystery (int x, int y, int z)
{
    int temp = x;
    if (y > temp) { temp = y; }
    if (z > temp) { temp = z; }
    return temp;
}
```

أ. اكتب ماذا يُعيد استدعاء العمليّة:
بلغة Java: mystery(7 , 9 , 9) ، أو بلغة C#: Mystery(7 , 9 , 9).

ب. ماذا تنفّذ العمليّة؟

/ يتبع في صفحة 3 /

2. أمامك الفئة **Rabbit** التي تمثّل أرنبًا. للفئة ثلاث صفات: الاسم – name ، الوزن بالكيلوغرامات – weight ، طول الأذنين بالسنتمترات – earLength.

الفئة مكتوبة بلغة Java وبلغة C# .

Java
```
public class Rabbit
{
    private String name;
    private double weight;
    private double earLength;
}
```

C#
```
public class Rabbit
{
    private string name;
    private double weight;
    private double earLength;
}
```

أ. اكتب بلغة Java أو بلغة C# عمليّة بنائيّة في الفئة **Rabbit**. تتلقّى العمليّة قيمة لكلّ صفة.

ب. افترض أنّه معطاة عمليّة رئيسيّة في الفئة Program. اكتب بلغة Java أو بلغة C# في العمليّة الرئيسيّة أمرًا لتكوين كائن rab1 من نمط **Rabbit** باسم Rodger ، وزنه – 30.0 كغم وطول أذنيه – 30.0 سم.

3. العددان المكوّنان من رقمين والأكبر من 0 ومركّبان من نفس الرقمين لكنّهما مختلفان عن بعضهما يُسمّيان **أخوين**.

أمثلة: العددان 13 وَ 31 هما **أخوان**.

للعدد 20 لا يوجد **أخ**، وللعدد 33 لا يوجد **أخ** .

اكتب بلغة Java أو بلغة C# عمليّة خارجيّة تتلقّى عددًا num صحيحًا ومكوّنًا من رقمين وأكبر من 0 .

تطبع العمليّة العدد الذي هو **أخ** العدد num. إذا لم يكن **أخ** للعدد، تطبع العمليّة البلاغ: 'No Brother'.

/ يتبع في صفحة 4 /

4. بغرض إدارة مكتبة في مدرسة ثانويّة معيّنة، تمّ تعريف فئة كتاب – **Book**.
صفات الفئة:

- اسم الكتاب – name من نمط نصّ.
- سنة الإصدار – year من نمط صحيح.
- مصادقة وزارة التربية والتعليم – permission من نمط بوليانيّ (true يشير إلى أنّ الكتاب مصادق عليه، false – الكتاب غير مصادق عليه).

لكلّ الصفات تمّ تعريف عمليّات set وَ get بلغة Java وعمليّات Set وَ Get بلغة C# .

عشيّة السنة الدراسيّة الجديدة، قرّر أمين المكتبة أن يُخرِج من المكتبة الكتب التي صدرت قبل سنة 2000 والكتب التي لم تحصل على مصادقة وزارة التربية والتعليم.

أ. اكتب بلغة Java أو بلغة C# عمليّة في الفئة **Book** تُعيد true – إذا كان يجب أن يبقى الكتاب في المكتبة، خلاف ذلك – تُعيد العمليّة false.

ب. اكتب بلغة Java أو بلغة C# عمليّة خارجيّة تتلقّى مصفوفة أحاديّة الأبعاد books من نمط **Book**، يمثّل كلّ واحد من حدودها كتابًا في المكتبة. تطبع العمليّة أسماء الكتب التي يجب إخراجها من المكتبة.
عليك استعمال العمليّة التي كتبتَها في البند "أ".

5. اكتب بلغة Java عمليّة خارجيّة باسم biggest أو بلغة C# عمليّة خارجيّة باسم Biggest.
تتلقّى العمليّة مصفوفة أحاديّة الأبعاد من نمط صحيح وتُعيد مؤشّر (إندِكس) الحدّ الذي قيمته هي الأكبر في المصفوفة.
إذا كانت القيمة الأكبر تظهر أكثر من مرّة واحدة في المصفوفة، تُعيد العمليّة مؤشّر المكان الأوّل في المصفوفة الذي تظهر فيه هذه القيمة.

/ يتبع في صفحة 5 /

الفصل الثاني (30 درجة)

أجب عن اثنين من الأسئلة 6-8 (لكلّ سؤال – 15 درجة).

6. للفئة فيلم سينمائيّ – **Movie** أربع صفات: اسم الفيلم – name من نمط نصّ، مدّة الفيلم بالدقائق – minutes من نمط صحيح، لغة الفيلم – lang من نمط نصّ، عدد الجوائز التي حاز عليها الفيلم – prizes من نمط صحيح.
لكلّ الصفات تمّ تعريف عمليّات get وَ set بلغة Java وعمليّات Get وَ Set بلغة C# ، كما وتمّ في الفئة تعريف عمليّة بنائيّة تتلقّى قيمة لكلّ صفة.

أ. اكتب بلغة Java أو بلغة C# عمليّة خارجيّة تتلقّى **فيلمَيْن سينمائيَّيْن** وتُعيد اسم الفيلم الذي حاز على عدد أكبر من الجوائز. إذا حاز الفيلمان على نفس عدد الجوائز – تُعيد العمليّة الكلمة "equals".

ب. أمامك قطعة من عمليّة رئيسيّة مكتوبة بلغة Java وبلغة C#.

Java
```
Movie m = new Movie("Love",120,"English", 5);
System.out.println(m);
```
C#
```
Movie m = new Movie("Love",120,"English", 5);
Console.WriteLine(m);
```

بعد تنفيذ قطعة العمليّة نتج المُخرَج التالي: **English/Love-120:5**
طبِّق العمليّة toString() بلغة Java أو ToString بلغة C# بحسب المُخرَج الذي نتج.

الفئة مهرجان – **Festival** تمثّل مهرجانًا يعرضون فيه أفلامًا مختلفة. للفئة ثلاث صفات: اسم المهرجان – name من نمط نصّ، والسنة التي أُجري فيها المهرجان – year من نمط صحيح، ومصفوفة أحاديّة الأبعاد – movies من نمط **Movie** .
لكلّ الصفات تمّ تعريف عمليّات get وَ set بلغة Java وعمليّات Get وَ Set بلغة C#.

تمّ في الفئة **Festival** تعريف عمليّة داخليّة بلغة Java exist(Movie mv)
وبلغة C# Exist(Movie mv) ، تفحص إذا كان الفيلم mv موجودًا في المهرجان.
إذا كان الفيلم موجودًا – تُعيد العمليّة true. خلاف ذلك – تُعيد العمليّة false.

جـ. اكتب عمليّة خارجيّة بلغة Java check(Festival f1, Festival f2)
أو بلغة C# Check(Festival f1, Festival f2).
تفحص العمليّة أنّه لا توجد أفلام مشتركة بين برنامجَي المهرجانين f1 وَ f 2. إذا لم يكن أيّ فيلم مشترك – تُعيد العمليّة true. خلاف ذلك – تُعيد العمليّة false.

/ يتبع في صفحة 6 /

7. أمامك قطعة من عمليّة رئيسيّة في الفئة Program مكتوبة بلغة Java وبلغة #C.
a وَ b هما مصفوفتان من نمط صحيح.

Java

```
int len = a.length;
int[] b = new int[len−1];
for (int k = 0; k< len−1; k++)
{
      if(k%2 == 0)
      {
            b[k] = a[k]+a[k+1];
      }
      else
      {
            b[k] = a[k]−a[k+1];
      }
}
```

C#

```
int len = a.Length;
int[] b = new int[len−1];
for (int k = 0; k < len−1; k++)
{
      if(k%2 == 0)
      {
            b[k] = a[k]+a[k+1];
      }
      else
      {
            b[k] = a[k]−a[k+1];
      }
}
```

/ يتبع في صفحة 7 /

أ. أمامك المصفوفة a .

	0	1	2	3	4	5	6
a	3	5	2	−9	6	4	0

تتبّع بواسطة جدول متابعة تنفيذ قطعة العمليّة. اعرض المصفوفة b بعد تنفيذ قطعة العمليّة.

يجب أن يشمل جدول المتابعة:

عمودًا لِـ k ، وعمودًا لِـ a[k] ، وعمودًا لِـ a[k+1] ، وعمودًا لِـ b[k] ، وعمودًا يُذكَر فيه إذا كان الشرط الذي في الأمر if يتحقّق أم لا يتحقّق.

ب. أعطِ مثالاً لمصفوفة أحاديّة الأبعاد a بِكبَر 7 تكون بالنسبة لها، بعد تنفيذ قطعة العمليّة، قيمة الحدّ الأوّل في المصفوفة b مساوية لقيمة الحدّ الأخير في المصفوفة b.

/ يتبع في صفحة 8 /

8. أجر العامل الجديد في مصنع معيّن هو 45 شيكلاً في الساعة. العامل الذي أقدميّته في العمل 5 سنوات أو أكثر يُعتبر عاملاً كبيرًا، وأجره هو 55 شيكلاً في الساعة.

أ. تمّ تعريف الفئة **Worker** ، التي صفاتها:

- رقم هويّة العامل – id من نمط صحيح.
- نوع العامل – senior من نمط بولينيّ (true – إذا كان العامل عاملاً كبيرًا، خلاف ذلك – false).
- عدد الساعات التي عملها في الشهر – numHours من نمط صحيح.

اكتب بلغة Java أو بلغة C# عنوان الفئة **Worker** وصفاتها.

ب. اكتب بلغة Java أو بلغة C# في الفئة **Worker** العمليّات التالية:

i عمليّة بنائيّة تتلقّى رقم هويّة وعدد سنوات الأقدميّة للعامل. تبتدئ العمليّة عدد الساعات التي عملها في الشهر لِـ 0 ، ونوع العامل بحسب عدد سنوات الأقدميّة.
في نهاية كلّ شهر يُجري المصنع تحديثًا لعدد ساعات العمل التي عملها العامل خلال الشهر.

ii اكتب عمليّة باسم salary بلغة Java أو Salary بلغة C# تحسب وتُعيد الراتب الشهريّ للعامل.

iii اكتب عمليّة باسم print بلغة Java أو Print بلغة C# تعرض كمُخرَج رقم الهويّة والراتب الشهريّ للعامل.

افترض أنّه تمّ لكلّ صفة تعريف عمليّات get وَ set بلغة Java وعمليّات Get وَ Set بلغة C# .

جـ. اكتب بلغة Java أو بلغة C# في الفئة Program عمليّة رئيسيّة تستقبل معطيات لِـ 251 عاملاً لشهر معيّن.
تعرض العمليّة الرئيسيّة كمُخرَج:

i بالنسبة لكلّ عامل رقم هويّته وراتبه الشهريّ.

ii معدّل الرواتب الشهريّة لجميع العاملين.

iii عدد العاملين غير الكبار الذين تقاضوا راتبًا شهريًّا أعلى من معدّل رواتب جميع العاملين.

/ يتبع في صفحة 9 /

الفصل الثالث (20 درجة)

أجب عن أحد السؤالين 9-10.

9. معطى عدد صحيح وأكبر من 0 .

خَلْط عدد هو عدد آخر بنفس الطول، مكوَّن من نفس الأرقام المكوَّن منها العدد الحاليّ، لكن بترتيب آخر. عدد مرّات ظهور كلّ رقم في العدد الحاليّ يساوي عدد مرّات ظهور نفس الرقم في العدد الآخر.

مثال: العدد 21611 .

خَلْطان ممكنان له هما العددان: 62111 ، 12116 .

معطاة الفئة **DigitStorage** التي تمثّل عددًا صحيحًا وأكبر من 0 . للفئة صفة واحدة:

مصفوفة count بِكبَر 10 من نمط صحيح. كلّ خليّة في المصفوفة تحوي عدد مرّات ظهور الرقم في العدد الذي قيمة هذا الرقم فيه مساوية لقيمة مؤشّر (إندِكس) الخليّة.

مثال: بالنسبة للعدد 21611 تبدو المصفوفة count على النحو التالي:

	0	1	2	3	4	5	6	7	8	9
count	0	3	1	0	0	0	1	0	0	0

أمامك جزء من الفئة **DigitStorage** مكتوب بلغة Java وبلغة C# :

```
public class DigitStorage
{
    private int[] count = new int[10];
}
```

أ. اكتب بلغة Java أو بلغة C#، في الفئة **DigitStorage** ، عمليّة بنائيّة تتلقّى عددًا صحيحًا وأكبر من 0 وتبتدئ المصفوفة count بحسب ذلك.

ب. اكتب بلغة Java أو بلغة C#، في الفئة **DigitStorage** ، عمليّة داخليّة تتلقّى كائنًا other من نمط **DigitStorage** . تُعيد العمليّة true – إذا كان الكائن other مطابقًا للكائن الحاليّ، خلاف ذلك– تُعيد العمليّة false.

جـ. اكتب بلغة Java أو بلغة C# عمليّة خارجيّة تتلقّى عددين صحيحين وأكبر من 0 ومختلفين. تُعيد العمليّة true – إذا كان العددان **خَلْطين**، خلاف ذلك – تُعيد العمليّة false. عليك استعمال العمليّتين اللتين كتبتَهما في البندين "أ" وَ "ب".

/ يتبع في صفحة 10 /

10. معطاة مصفوفة ثنائيّة الأبعاد a تحوي أعدادًا صحيحة مختلفة.

نعرّف **عدد السوابق** للحدّ (i, j) في المصفوفة على النحو التالي:

عدد الأعداد الواقعة قبله في المصفوفة عندما نمسح المصفوفة بالتسلسل (حسب الأسطر) ابتداءً من المكان (0, 0) وحتّى المكان (i, j) (لا يشمل (i, j)).

مثال: بالنسبة للمصفوفة التي بِكبَر 7×5 التي أمامك:

39	72	3	8	0	2	54
18	16	**13**	47	41	87	65
34	78	37	99	97	46	56
51	50	12	31	58	63	18
23	4	11	26	83	25	67

عدد سوابق العدد 39 هو 0، لأنّه لا توجد قبله أعداد في المصفوفة.

عدد سوابق العدد 13 هو 9، لأنّه توجد قبله 9 أعداد في المصفوفة:

39, 72, 3, 8, 0, 2, 54, 18, 16.

أ. اكتب بلغة Java عمليّة place أو بلغة C# عمليّة Place تتلقّى مصفوفة ثنائيّة الأبعاد لأعداد صحيحة مختلفة وعددًا صحيحًا x. تُعيد العمليّة **عدد سوابق** الحدّ الذي قيمته هي x.

إذا لم يكن x موجودًا في المصفوفة، تُعيد العمليّة 1−.

(انتبه: تكملة السؤال في الصفحة التالية.)

/ يتبع في صفحة 11 /

ب. اكتب بلغة Java عمليّة printAndCount أو بلغة C# عمليّة PrintAndCount، تتلقّى مصفوفة ثنائيّة الأبعاد لأعداد صحيحة مختلفة، وزوج أعداد إضافيًّا يشير إلى **العددين السابقين** لحدّين في المصفوفة. العدد الأوّل المتلقَّى هو أصغر من العدد الثاني.
يجب على العمليّة أن تطبع جميع الحدود التي في المجال الذي بين الحدّين اللذين تمّ تلقّي **عدديهما السابقين** (بما في ذلك الحدّان).
مثال بالنسبة للمصفوفة التي بِكِبَر 7×5 التي أمامك و**العددان السابقان** 19 وَ 29 :

39	72	3	8	0	2	54
18	16	13	47	41	87	65
34	78	37	99	97	46	56
51	50	12	31	58	63	18
23	4	11	26	83	25	65

المُخرَج هو (من اليسار إلى اليمين): 4 , 23 , 18 , 63 , 58 , 31 , 12 , 50 , 51 , 56 , 46 .

בהצלחה!
نتمنّى لك النّجاح!

זכות היוצרים שמורה למדינת ישראל.
אין להעתיק או לפרסם אלא ברשות משרד החינוך.
حقوق الطّبع محفوظة لدولة إسرائيل.
النّسخ أو النّشر ممنوعان إلّا بإذن من وزارة التّربية والتّعليم.