מדינת ישראל
משרד החינוך
סוג הבחינה: בגרות
מועד הבחינה: קיץ תשע"ז, 2017
מספר השאלון: 899381
תרגום לאמהרית (6)

מדעי המחשב

על פי תכנית הרפורמה ללמידה משמעותית

הוראות לנבחן

א. משך הבחינה: שלוש שעות.

ב. מבנה השאלון ומפתח ההערכה:
בשאלון זה שלושה פרקים.
פרק ראשון – בפרק זה שלוש שאלות, ענה על פי ההוראות בפרק.
(1×10) + (1×15) — 25 נקודות
פרק שני – בפרק זה יש שלוש שאלות, ומהן עליך לענות על שתיים.
(2×25) — 50 נקודות
פרק שלישי – בפרק זה שאלות בארבעה מסלולים שונים. ענה על שאלה אחת במסלול שלמדת.
(1×25) — 25 נקודות
סה"כ — 100 נקודות

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

ד. הוראות מיוחדות:
(1) את כל התכניות שאתה נדרש לכתוב בשפת מחשב בפרק הראשון כתוב בשפה אחת בלבד — Java או C#.
(2) **רשום על הכריכה החיצונית** של המחברת באיזו **שפה** אתה כותב — Java או C#.
(3) **רשום על הכריכה החיצונית** של המחברת את שם **המסלול שלמדת.**
המסלול הוא אחד מארבעת המסלולים האלה:
מערכות מחשב ואסמבלי, מבוא לחקר ביצועים, מודלים חישוביים, תכנות מונחה עצמים.
הערה: בתכניות שאתה כותב לא יורדו לך נקודות, אם תכתוב אות גדולה במקום אות קטנה או להפך.

እስራኤል አገር
የትምህርት ሚኒስቴር
የፈተናው ዓይነት፦ የ2ኛ ደረጃ ት/ቤት መልቀቂያ ፈተና
የፈተናው ጊዜ፦ የ2017 ዓ.ም. የበጋ ፈተና
የፈተናው ቁጥር፦ 899381
ወደ አማርኛ የተተረጎመ (6)

የኮምፒዩተር ሳይንስ

በሪፎርሙ ትርጉም ያለው የማስተማር ዕቅድ መሠረት

ለተፈታኙ የቀረቡ መመሪያዎች፦

ሀ. ለፈተናው የተሰጠው ጊዜ፦ ሦስት ሰዓት።

ለ. የፈተናው መዋቅርና የነጥብ አሰጣጥ፦ በዚህ መጠይቅ ሦስት ምዕራፎች አሉ።
ምዕራፍ አንድ - በዚህ ክፍል ውስጥ ሦስት ጥያቄዎች አሉ፣ በክፍሉ በተሰጡት መመሪያዎች መሠረት መልስ።
(1 × 10) + (1 × 15) — 25 ነጥብ
ምዕራፍ ሁለት - በዚህ ክፍል ካሉት ሦስት ጥያቄዎች ውስጥ ሁለቱን መልስ።
(2×25) — 50 ነጥብ
ምዕራፍ ሦስት - በዚህ ክፍል በአራት የተለያዩ መንገዶች የቀረቡ ጥያቄዎች አሉ፣ አንድ ጥያቄ በተማርከው መንገድ መልስ።
(1 × 25) — 25 ነጥብ
በአጠቃላይ — 100 ነጥብ

ሐ. ለመጠቀም የተፈቀደ የመርጃ መሣሪያ፦
ፕሮግራም መደረግ ከሚችል ኮምፒውተር በስተቀር (מחשב הניתן לתכנות) ሁሉንም የመረጃ መሣሪያዎች መጠቀም ይቻላል።

መ. ልዩ መመሪያዎች፦
(1) በክፍል አንድ እንድትፅፍ የተጠየካቸውን የኮምፒዩተር ቋንቋ ፕሮግራሞች በሙሉ በአንድ ቋንቋ ብቻ ፃፍ፣ በC# ወይም በJava።
(2) ለመፃፍ የምትጠቀመውን **ቋንቋ፦** C# ወይም Java፣ በደብተሩ **ሽፋን ላይ በውጪ በኩል ጻፍ።**
(3) በየትኛው **መንገድ እንደተማርክ፦** በደብተሩ **ሽፋን ላይ በውጪ በኩል ጻፍ።**
መንገዱም ከሚከተሉት አራት መንገዶች አንዱ ነው።
የኮምፒዩተር ሲስተም እና አሴምብለር÷ ተግባር ላይ የማዋል ጥናት መሠረታዊ ት/ት÷ የስሌት ሞዴሎች÷ በአካሎች የሚመራ ፕሮግራሚንግ።
ማሳሰቢያ፦ በምትጽፋቸው ፕሮግራሞች ውስጥ በትንሽ ፊደል ፋንታ ትልቅ ፊደል ወይም በተቃራኒው ብትጽፍ ነጥብ አይቀነስብህም።

כתוב במחברת הבחינה בלבד, בעמודים נפרדים, כל מה שברצונך לכתוב כטיוטה (ראשי פרקים, חישובים וכדומה) רשום "טיוטה" בראש כל עמוד טיוטה רישום טיוטות לשהן על דפים מחוץ למחברת הבחינה עלול לגרום לפסילת הבחינה

እንደ ንድፍ (טיוטה) መፃፍ የምትፈልገውን በሙሉ (አርዕስቶች፣ ስሌቶች፣ ወዘተ) በፈተናው ደብተር ላይ ብቻ ጻፍ፣ እንዲሁም በተለዩ ገፆች ላይ ጻፍ። በእያንዳንዱ የንድፍ ገጽ ላይ "ንድፍ" ብለህ ጻፍ። ንድፎችን በሌላ ወረቀት ላይ ስትጽፍ ከተገኘህ ፈተናህ ውድቅ ሊሆን ይችላል።

ההנחיות בשאלון זה מנוסחות בלשון זכר ומכוונות לנבחנות ולנבחנים כאחד.

በዚህ ፈተና ውስጥ የቀረቡት መመሪያዎች በወንድ ጾታ የቀረቡ ቢሆኑም ለሁለቱም ፆታዎች ያገለግላሉ።

בהצלחה!

መልካም ዕድል!

ጥያቄዎች

ይህ ፈተና ሦስት ክፍሎች አሉት።

ለእያንዳንዱ ክፍል በተሰጠው መመሪያ መሠረት ከሦስቱም ክፍሎች ጥያቄዎችን መመለስ አለብህ።

ክፍል አንድ (25 ነጥብ)

ማስገንዘቢያ፦ ኢንፑት/መቀበል (קלט) በተጠየክበት ጥያቄ ሁሉ የኢንፑቱን ትክክለኛነት መመርመር አስፈላጊነት የለውም።

በJava ለሚሠሩ፦ መቀበል (ኢንፑት) በሚያስፈልግበት ጥያቄ ሁሉ በፕሮግራሙ የሚከተለው ትዕዛዝ መኖሩን አስብ፦

Scanner input = new Scanner (System.in);

ጥያቄ 1 ን መልስ – ግዴታ ነው (10 ነጥብ)።

1. የሙሉ ቁጥሮችን ድርድር የሚቀበል ውጫዊ ስልት big ን በJava ወይም Bigን በC# ጻፍ። ስልቱ በድርድሩ ውስጥ ዋጋው እጅግ ከፍተኛ የሆነውን የአካል ጠቃሽ (ኢንዴክስ) ይመልስ/ያሳይ።
   ቁጥሩ በድርድሩ ውስጥ ከአንድ ጊዜ በላይ ከታየ/ከቀረበ÷ ስልቱ ከቀረቡት/ከታዩት ጠቃሾች (ኢንዴክሶች) መካከል በጣም አነስተኛውን ይመልስ/ያሳይ።

ከ2-3 ካሉት ጥያቄዎች ውስጥ አንዱን መልስ (ለጥያቄው – 15 ነጥብ)።

2. ለዓለም አቀፍ የስፖርት ጨዋታ ዋዜማ ዝግጅት ሦስት ይዘቶች (תכונות) ያለውን ክፍል **Game** ወሰኑ (הגדירו)÷ ይዘቶቹም፦ ሡንሰለት ዓይነት የሆነ የጫወታው ስም – gameName ÷ ሙሉቁጥር ዓይነት የሆነ የተጫዋቾቹ ብዛት/ቁጥር – numPlayers ÷ ጫወታው በውሃ ውስጥ የሚካሄድ ከሆነ "ቦሊያን" ዓይነት የሆነ – isWater የሚሉት ናቸው።
   ለእያንዳንዱ ይዘት በJava ስልቶች get እና set በ C# ስልቶች Get እና Set ተወስነዋል ብለህ አስብ።
   - ኦ. ለእያንዳንዱ ይዘት ዋጋዎችን የሚቀበል ገንቢ ስልት በክፍል **Game** በJava ወይም በC# ጻፍ።
   - ኀ. ሁለት ይዘቶች ያሉት ክፍል ÷ ሀገር — **Country** ን ወሰኑ (הגדירו)፦ መለያ/ይዘቶቹም፦ ሡንሰለት ዓይነት የሆነ የሀገሪቱ ስም – countryName ÷ **Game** ዓይነት መጠኑ 43 የሆነ ባለአንድ ዳይሜንሽናል ድርድር – games የሚሉት ናቸው። ድርድሩ ሀገሪቷ የምትሳተፍባቸውን ጫወታዎች ይይዛል። ለእያንዳንዱ ይዘት በJava ስልቶች get እና set በ C# ስልቶች Get እና Set ተወስነዋል ብለህ አስብ።
     - (1) የክፍል **Country**ን ርዕስ እና ይዘቶቹን በJava ወይም በC# ጻፍ።
     - (2) የሀገር ስም የሚቀበል ገንቢ ስልት በክፍል **Country** በJava ወይም በC# ጻፍ።
       ስልቱ በአንድም ጫወታ ለማትሳተፍ ሀገር አካል (מצע) ይገንባ።
     - (3) የጫወታውን ስም የሚቀበልና ሀገሪቱ በጫወታ የምትሳተፍ ከሆነ true — የሚመልስ አለበለዚያ falseን ይመልስ ዘንድ ቦሊያን ስልት የሆነ በክፍል **Country** በJava ወይም በC# ጻፍ።

/ገጽ 3 ላይ ይቀጥላል/

3. በJava እና በC# የተጻፈ የፕሮግራም ክፍል በፊትህ አለ።

Java

```java
int i , j;
for (i = 1; i < = n; i++)
{
    a[i] = i;
}
for (i = 2; i < n; i++)
{
    if (a[i] != 0)
    {
        for ( j = i + 1; j < = n; j++)
        {
            if (a[ j] % a[i] = = 0)
            {
                a[ j] = 0;
            }
        }
    }
}

for (i = 1; i < = n; i++)
{
    if (a[i] != 0)
    {
        System.out.println(a[i]);
    }
}
```

C#

```csharp
int i , j;
for (i = 1; i < = n; i++)
{
    a[i] = i;
}
for (i = 2; i < n; i++)
{
    if (a[i] != 0)
    {
        for ( j = i + 1; j < = n; j++)
        {
            if (a[ j] % a[i] = = 0)
            {
                a[ j] = 0;
            }
        }
    }
}

for (i = 1; i < = n; i++)
{
    if (a[i] != 0)
    {
        Console.Writeln(a[i]);
    }
}
```

ሀ. ለ$n = 15$ ምን እንደሚታተም ጻፍ።

ለ. የፕሮግራሙ ክፍል ምን እንደሚፈጽም በአንድ ዐረፍተ-ነገር ጻፍ፣ ማለትም መፍትሔው የፕሮግራሙ ክፍል የሆነን ፕሮብሌም ቀምር/ጻፍ።

/ገጽ 4 ላይ ይቀጥላል/

ክፍል ሁለት (50 ነጥብ)

ልብ በል፦ ኮዱን ጽፈህ እንድትተገብር በተጠየቅባቸው ጥያቄዎች ሁሉ ኮዳቸውን ጽፈህ ሳትተገብር በሚከተሉት ክፍሎች መጠቀም ትችላለህ፦ ተራ፣ ካርታ፣ ባይናሪ ዛፍ እና ዘርፍ። በተጨማሪ ስልቶች (פעולות) የምትጠቀም ከሆነ ኮዳቸውን ጽፈህ መተግበር ይኖርብሃል።

ከ4-6 ካሉት ጥያቄዎች ውስጥ ሁለቱን መልስ (እያንዳንዱ ጥያቄ 25 ነጥብ)።

4. በሆነ የመረጃዎች አወቃቀር የሚሠሩ አምስት ድርጊቶች ውሳኔ በፊትህ አለ።
ልብ በል፦ በፊትህ ያሉት የድርጊቶች ስሞች በJava ወይም በC# የተጻፉ አይደሉም።

insert(x) — ዋጋው x የሆነን ሙሉ ቁጥር ዓይነት አካል ወደመዋቅሩ የሚያስገባ ድርጊት።

showMin() — መዋቅሩን ሳይለውጥ በመዋቅሩ ውስጥ ዝቅተኛውን ዋጋ የሚመልስ ድርጊት።

getMax() — በመዋቅሩ ውስጥ ከፍተኛ ዋጋ ያለውን አካል የሚመልስ እና ከመዋቅሩ ውስጥ የሚያወጣው ድርጊት። ከአንድ በላይ እንደዚህ ዓይነት አካል ካለ ድርጊቱ መጀመሪያ የሚከሰተውን ያስገባና ያውጣ።

exists(x) — ዋጋው x የሆነው አካል በመዋቅሩ ውስጥ ካለ trueን አለበለዚያ ግን falseን የሚመልስ/የሚያሳይ ቦሊያኒክ ድርጊት።

div7() — ዋጋው ለ7 ያለቀሪ የሚካፈል አካል በመዋቅሩ ውስጥ ካለ trueን አለበለዚያ ግን falseን የሚመልስ/የሚያሳይ ቦሊያኒክ ድርጊት።

ከዚህ በላይ ከተወሰኑት አምስት ድርጊቶች ውስጥ የተለያዩ ውስብስቦችን ለመተግበር መስፈርት የሚያሟሉትን መረጃዎች መዋቅር ለማቅረብ እንፈልጋለን።

ለምሳሌ፦ ድርጊቶች insert , showMin ን በውስብስብነት (סיבוכיות) O(1) እና ድርጊቶች getMax, exists ን በውስብስብነት (סיבוכיות) O(n) በላዩ ለመፈጸም የሚቻል የመረጃዎች መዋቅር ለማቅረብ ይፈለጋል። ለዚህም የመረጃዎችን መዋቅር እንወስን (יגדנו) እና ድርጊቶቹ እንዴት እንደሚተገበሩ እናብራራለን።

ልብ በል፦ በዚህ መዋቅር ለድርጊት div7 ትኩረት መስጠት አያስፈልግም።

/ገጽ 5 ላይ ይቀጥላል/

ተስማሚው የመረጃዎች መዋቅር የተዋቀረው ከ፦

- ሙሉ ቁጥር lst የሆነ፣ የተያያዘ የሁለትዮሽ አቅጣጫ (רשימה מקושרת דו־כיוונית) መዝገብ።
- ወደመዝገቡ ውስጥ የገባ ሚኒማል አካል ጠቋሚ፣ min።

ስልቶቹ እንደሚከተለው ይከናወናሉ፦

ስልቱ (Method)	እንዴት እንደሚተገበር አብራራ	ትግበራው ለምን ውስብስብነትን መቋቋም እንደሚችል ማስረጃ
insert(x)	– አካል x ን በመዝገቡ መጀመሪያ (ራስ) ለማስገባት። – አካሉ እስካሁን ከሚኒመሙ የሚያንስ ከሆነ፣ ሪፖርት የሚጠቁመው min አዲሱን አካል ይጠቁማል።	– አካልን በመዝገቡ መጀመሪያ ላይ – O(1) ማስገባት። – ከሚኒመሙ የሚያንስ አካል ገብቶ ከሆነ፣ ሪፖርት ጠቋሚው ወደ ሚኒማል አካሉ – O(1) ይሆናል። – የመጀመሪያው አካል ገብቶ ከሆነ – በመዝገቡ መጀመሪያ ላይ O(1) ማስገባት እና ሪፖርት ጠቋሚው ሚኒማል ወደ ሆነው አንደኛው አካል። በአጠቃላይ – O(1)።
showMin()	min የሚያመለክተውን አካል ዋጋ መመለስ።	ዋጋ – O(1) ን መመለስ።
exists(x)	በመዝገብ lst ላይ ማለፍና ዋጋው x የሆነውን አካል መፈለግ።	ከፋ ቢባል – በሙሉ መዝገቡ ላይ ማለፍ – O(n)።
getMax()	በመዝገብ lst ላይ ማለፍ፣ ዋጋው ማክሲማል የሆነውን አካል መፈለግ እና ከመዝገቡ ውስጥ ማውጣት።	– ዋጋው ማክሲማል O(n) የሆነውን አካል መፈለግ። – ከመዝገብ O(1) ውስጥ መውጣቱ።

በፊትህ <u>ሁለት</u> ንውሳን א - ב ይገኛሉ። ከንውሳኑ በእያንዳንዳቸው የተዘረዘሩትን መስፈርቶች የሚያሟላ የመረጃዎች መዋቅር ማቅረብ ይኖርብሃል።
መዋቅሩ ከተማርካቸው የተወሰኑ መዋቅሮች እና ዓይነቶች የተዋቀረ ሊሆን ይችላል።
<u>ለእያንዳንዱ</u> ስልት እንዴት እንደምትተገብረው አብራራ እና ትግበራው በምሳሌው ሠንጠረዥ እንደተሰጠው ለምን የሚጠየቀውን መስፈርት እንደሚያሟላ አስረዳ። ስልቱን መተግበር (ኮዱን መጻፍ) <u>አያስፈልግም</u>።

א. ስልቶች insert , existsን በውስብስብነት O(n) መተግበር እና ስልቶች showMin , getMax ን በውስብስብነት O(1) መተግበር።

ב. ስልቶች insert , getMax ን በውስብስብነት O(n) መተግበር እና ስልት div7ን በውስብስብነት O(1) መተግበር።

/ገጽ 6 ላይ ይቀጥላል/

5. በፊትህ በJava እና በC# የተጻፈ ፕሮግራም አለ።
ስልት what በJava እና What በ C# ሙሉ ቁጥሮችን የያዘ ባለ አንድ ዳይሜንሽናል ድርድር እና ሙሉ ቁጥር k፣ k > 0 ን ይቀበላል። በድርድሩ ያሉ ቁጥሮች በሙሉ በ0 እና በk (ጨምሮ) መካከል ያሉ ቁጥሮች ናቸው።

Java

```
public class Program
{
    public static int[]  what(int[] arr, int k)
    {
        int n = arr.length;
        int[] b = new int[n];
        int[] c = new int[k+1];

        for (int i = 0; i < k+1; i++) c[i] = 0;
        for (int j = 0; j < n; j++)  c[arr[ j]] = c[arr[ j]] + 1;      //*

        for (int i = 1; i < k+1 ; i++) c[i] = c[i] + c[i−1];        //**

        for (int j = n−1 ; j >= 0; j--)                              //***
        {
            b[c[arr[ j]] − 1] = arr[ j];
            c[arr[ j]] = c[arr[ j]] −1;
        }
        return b;
    }


    public static void main(String[] args)
    {
        int[] arr = new int[] {5,0,2,1,3,0,5};
        arr = what(arr, 5);
        for (int i = 0; i < arr.length; i++) System.out.println(arr[i]);
    }
}
```

/ገጽ 7 ላይ ይቀጥላል/

C#

```
public class Program
{
      public static int[]  What(int[] arr, int k)
      {
         int n = arr.Length;
         int[] b = new int[n];
         int[] c = new int[k+1];

         for (int i = 0; i < k+1; i++) c[i] = 0;
         for (int j = 0; j< n; j++)  c[arr[ j]] = c[arr[ j]] + 1;    //*

         for (int i = 1; i< k+1 ; i++) c[i] = c[i] + c[i−1];      //**

         for (int j = n−1 ; j >= 0; j--)                          //***
         {
            b[c[arr[ j]] −1] = arr[ j];
            c[arr[ j]] = c[arr[ j]] −1;
         }
         return b;
      }

      public static void Main(string[] args)
      {
            int[] arr = new int[] {5,0,2,1,3,0,5};
            arr = What(arr, 5);
            for (int i = 0; i < arr.Length; i++) Console.WriteLine(arr[i]);
      }
}
```

ሀ. (1) በ//* የተመለከተውን "ዓምድ" (לולאה) ከፈጸምክ በኋላ ድርድር c ን ንደፍ/ሣል።

(2) በ//** የተመለከተውን "ዓምድ" ከፈጸምክ በኋላ ድርድር c ን ንደፍ/ሣል።

(3) በ//*** የተመለከተውን "ዓምድ" አፈጻጸም ተከታተል።
በክትትሉ j , arr[j] , c[arr[j]] , b[c[arr[j]] − 1] , ድርድር b ን እና ድርድር cን ማሳየት አለብህ።

ለ. ስልት what በJava ወይም What በ C# ምን ይፈጽማል?

ሐ. የስልት what በJava ወይም What በ C# የማስሮጥ ጊዜ ውስብስብነት ምንድን ነው? መልስህን አስረዳ።

/ገጽ 8 ላይ ይቀጥላል/

6. ሀ. ውጫዊ ስልት existን በJava ወይም Existን በC# ተግብር።
ስልቱ መሉና ሙሉ ቁጥር x ዓይነት የሆነ ባይናሪ ዛፍ t ይቀበላል። ዛፉ ዋጋው x የሆነ መገናኛ (צומת) ካለው ስልቱ true ይመልሳል/ያሳያል፣ አለበለዚያ፦ — ስልቱ false ይመልስ። ዛፉ ባዶ ከሆነ — ስልቱ false ይመልስ።

ለ. በፊትህ በJava፦ check(t1, t2) እና በC#፦ Check(t1, t2) የሆነ ስልት አለ።
ስልቱ ባዶ ያልሆኑ ሙሉ ዓይነት t1 እና t2 ሁለት ባይናሪ ዛፎች የሚቀበል እና በዛፍ t1 የሚገኙ እና በዛፍ t2 <u>የማይገኙ</u> ቁጥሮችን በሙሉ የያዘ መዝገብ የሚመልስ ነው። እንዲሁም ስልቱ ሦስት ፓራሜትሮችን የሚቀበል ተጨማሪ ስልት ያዛል።

<u>Java</u>

```
public static Node<Integer> check(BinNode<Integer> t1, BinNode<Integer> t2)
{
    Node<Integer> first = new Node<Integer> (−1);
    first = check(t1 , t2 , first);
    return first.getNext();
}
```

<u>C#</u>

```
public static Node<int> Check(BinNode<int> t1 , BinNode<int> t2)
{
    Node<int> first = new Node<int> (−1);
    first = Check(t1 , t2 , first);
    return first.GetNext();
}
```

ስልቱን ኮዱን በመጻፍ ተግብር (ממש)

በJava፦

```
public static Node<Integer> check( BinNode<Integer> t1,
                                   BinNode<Integer> t2 , Node<Integer> list)
```

ወይም በC#፦

```
public static Node<int> Check(BinNode<int> t1,BinNode<int> t2 ,Node<int> list)
```

በንዑስ ሀ በተገበርከው ስልት መጠቀም ትችላለህ።

ሐ. በንዑስ ለ የተገበርከው ስልት የማስሮጥ ጊዜ ውስብስብነት ምንድን ነው?
መልስህን አስረዳ።

/ገጽ 9 ላይ ይቀጥላል/

ክፍል ሦስት (25 ነጥብ)

በዚህ ክፍል ውስጥ በአራት የትምህርት መንገዶች (ማስሉሊም) የተዘጋጁ ጥያቄዎች አሉ፦

የኮምፒዩተር ሥርዓቶች እና አሴምብሊይ ከገጽ 9 - 11

ተግባር ላይ የማዋል ጥናት መሠረታዊ ትምህርት (מבוא לחקר ביצועים) ከገጽ 12 - 16

የስሌት ሞዴሎች (מודלים חישוביים) ገጽ 17

በአካሎች የሚመራ ፕሮግራሚንግ (תכנות מונחה עצמים) በ Java ፤ ከገጽ 18 - 21 በአካሎች የሚመራ ፕሮግራሚንግ በ C# ፤ ከገጽ 22 - 25

በተማርከው መንገድ አንድ ጥያቄ መልስ።

የኮምፒዩተር ሥርዓቶች እና አሴምብሊይ

በዚህ መንገድ የተማርክ ከሆነ ከ7-8 ካሉት ውስጥ አንድ ጥያቄ መልስ (25 ነጥብ)።

7. በዚህ ጥያቄ ውስጥ ሁለት ንዑሳን ጥያቄዎች א-ב አሉ። በንዑሳኑ መካከል ግንኙነት የለም። ሁለቱንም መመለስ ይገባሃል።

   א. በAL እና በ AH አንዱ ከአንዱ የተለዩ ምልክት ያልተደረገባቸው ቁጥሮች ያሉበት በአሴምብሊይ ቋንቋ የተጻፈ የፕሮግራም ክፍል በፊትህ ይገኛል።

```
       MOV   AX , 7F80H
       CMP   AH , AL
       JL    SMALL
BIG:   MOV   BL , AH
       JMP   SOF
SMALL: MOV   BL , AL
SOF:
```

   (1) በፕሮግራሙ ክፍል አፈጻጸም መጨረሻ ላይ የBL ዋጋ ስንት ይሆናል?

   (2) የፕሮግራሙ ክፍል ምን ይፈጽማል?

   (3) በፕሮግራሙ ክፍል በትዕዛዝ JL SMALL ፋንታ ትዕዛዝ JB SMALL ተጻፈ። በፕሮግራሙ ክፍል አፈጻጸም መጨረሻ ላይ ትዕዛዙ ከተቀየረ በኋላ የBL ዋጋ ከትዕዛዙ መቀየር በፊት ከነበረው ዋጋው የተለየ ይሆናልን? መልስህን አብራራ።

/ገጽ 10 ላይ ይቀጥላል/

ጋ. (ከንዑስ ጥያቄ ለ ጋር ግንኙነት የለውም።)
በፊትህ ስድስት ዐረፍተ-ነገሮች አሉ ለእያንዳንዳቸው እውነት ወይም ሐሰት መሆኑን ወስን። ዐረፍተ-ነገሩ ሐሰት ከሆነ ለምን እንደሆነ አብራራ።

(1) $10011110101110_2 > 27AE_{16}$

(2) $27AE_{16} > 10159_{10}$

(3) የማጠራቀሚያዎች BP እና SP የሥራ ድርሻዎች፦
BP አንድ ቃል በካርታው (המחסנית) ውስጥ በየትኛው አድራሻ መጻፍ እንዳለበት እና ቃል ከየትኛው አድራሻ መመዘዝ እንዳለበት ይወስናል።
SP ያለ BP ጣልቃገብነት÷ ቅደም ተከተሉ (פרוצדורה/שגרה) ፓራሜትሮችን ከካርታው (מחסנית) ውስጥ እንዲመዝ ያስችለዋል።

(4) በንባቡ መጨረሻ ላይ እና በሁሉም ትዕዛዛት መፈጸሚያ ላይ የ IP ዋጋ ሁል ጊዜ በ1 ይጨምራል።

(5) በፊትህ ሁለት የአሴምብሊ የፕሮግራም ክፍሎች አሉ።

ክፍል 2	ክፍል 1
MOV DX , 0000H	MOV DX , 0000H
MOV AX , 0064H	MOV AX , 0064H
DIV AX	DIV AL

በሁለቱ ክፍሎች አፈጻጸም መጨረሻ ላይ የAX ይዘት 0001H ይሆናል።

(6) በፊትህ ሁለት የአሴምብሊ የፕሮግራም ክፍሎች አሉ።

ክፍል 2	ክፍል 1
MOV AL , 5BH	MOV AL , 5BH
MOV CL , 9	MOV CL , 8
ROL AL , CL	ROL AL , CL

በክፍል 1 አፈጻጸም መጨረሻ ላይ የAL ዋጋ በመጀመሪያው ትዕዛዝ ከተቀመጠበት ዋጋ ጋር እኩል ይሆናል፣ እንደዚሁም በክፍል 2 አፈጻጸም መጨረሻ ላይ የAL ዋጋ በመጀመሪያው ትዕዛዝ ከተቀመጠበት ዋጋ ጋር እኩል ይሆናል።

/ገጽ 11 ላይ ይቀጥላል/

8. ሁለት ድርድሮች ARR1 እና ARR2 ተሰጥተዋል። ሁለቱ ድርድሮች እኩል መጠን ያላቸው ሲሆን እያንዳንዱ አካላቸው የአንድ ቃል መጠን አለው። በእያንዳንዱ ድርድር ያለው እያንዳንዱ አካል፥ በክልል FFFFH - 1000H ባለ አራት ዲጂት ሙሉ ሄክሳዴሲማል ቁጥር ተቀምጦበታል።

እንደዚሁም የቤት መጠን ያለው፣ ARR1 እና ARR2 የሆነ መጠን የተቀመጠበት ተለዋጭ K ተሰጥቷል።

የቁጥሮችን ጥንድ ቁጥሮች (ከARR1 ድርድር አንዱን እና ከARR2 ድርድር አንዱን) በማጠራቀሚያ DX የሚያጠራቅም/የሚያስቀምጥ እና የሚከተሉትን መስፈርቶች የሚያሟላ የፕሮግራም ክፍል በአሴምብሊ ጻፍ፦

(i) በድርድሩ ተጣምደው ላሉ ሁለት ቁጥሮች አንድ አመልካች (ኢንዴክስ) ነው ያላቸው።

(ii) ተጣምደው ያሉት ሁለቱ ቁጥሮች የተዋቀሩት ከአንድ ዓይነት ዲጂቶች ሲሆን ቅደም ተከተላቸው ግን በተቃራኒው ነው።

<u>ምሳሌ</u>፦ መጠናቸው 5 ለሆነ በፊትህ ላሉ ድርድሮች ARR1 እና ARR2 በማጠራቀሚያ DX ቁጥር 2 ይቀመጣል።

ARR1	2025H	1061H	1492H	5777H	1948H
ARR2	1984H	1601H	2914H	9999H	8491H

/ገጽ 12 ላይ ይቀጥላል/

ተግባር ላይ የማዋል ጥናት መሠረታዊ ትምህርት

በዚህ መንገድ የተማርክ ከሆነ ከ9-10 ካሉት ጥያቄዎች ውስጥ አንድ ጥያቄ መልስ (25 ነጥብ)።

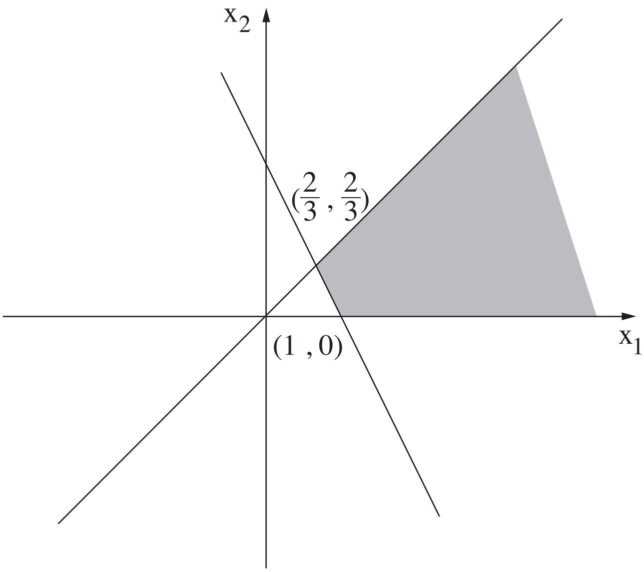
9. ሊኒያር የፕሮግራሚንግ (תכנון) ችግር (ፕሮብሌም) ተሰጥቷል።

$$\max\{z = 5x_1 - x_2\}$$

ቀጥለው ባሉት ገዴታዎች (אילוצים) ሥር፦

$$x_2 \le x_1$$
$$2x_1 + x_2 \ge 2$$
$$x_1 \ge 0$$
$$x_2 \ge 0$$

ለተሰጠው ችግር፥ መፍትሔዎች ሊሆኑ የሚችሉ፥ ንድፍ/ሥዕል ክልል በፊትህ ቀርቧል።



በሚቀጥለው ገጽ ላይ ከ א - ה ያሉት ንዑሳን እያንዳንዳቸው የተሰጠውን ሊኒያር የፕሮግራሚንግ ችግር (ፕሮብሌም) ይመለከታሉ። ንዑሳኑ א - ה አንዱ ከአንዱ ጋር ግንኙነት የላቸውም። በሁሉም ንዑሳን ጥያቄዎች ላይ መልስ ስጥ።

ከ א - ד ላሉት ንዑሳን ከi- iv ያሉ አራት መግለጫዎች በፊትህ ተሰጥተዋል፦
ለእያንዳንዱ ንዑስ አንዱ መግለጫ ብቻ ነው ትክክል።

i ኦፕቲማል መፍትሔው አንድ ነው።

ii እልቀቢስ ኦፕቲማል መፍትሔዎች ናቸው ያሉት።

iii ኦፕቲማል መፍትሔው አልታገደም (לא חסום)።

iv ኦፕቲማል መፍትሔ የለም።

/ገጽ 13 ላይ ይቀጥላል/

ከ א - ד ላሉት ንውሳን ለእያንዳንዳቸው የትኛው መግለጫ ትክክል እንደሆነ ወስን።
ወደ ደብተርህ ገልብጠውና ውሳኔህን አስረዳ።

- ለሆነ ንዑስ መግለጫ i ን ከመረጥክ ÷ ያለውን ብቸኛ ኦፕቲማል መፍትሔ እና ባገኘኸው መፍትሔ የዓላማውን ፈንክሽን ዋጋ መመዝገብ አለብህ።
- ለሆነ ንዑስ መግለጫ ii ን ከመረጥክ ÷ ለችግሩ አጠቃላዩን ኦፕቲማል መፍትሔ እና በኦፕቲማል መፍትሔዎቹ ክልል የዓላማውን ፈንክሽን ዋጋ ማግኘት አለብህ።

א. በጥያቄው መጀመሪያ ለተሰጠው ሊኒያር የፕሮግራሚንግ ችግር የትኛው መግለጫ ነው ትክክል? መልስህን አስረዳ።

ב. በጥያቄው መጀመሪያ ላይ የተሰጠውን ችግር የዓላማ ፈንክሽን ብቻ ወደ

$\max\{z = -5x_1 + x_2\}$ ይቀይሩታል።

የዓላማው ፈንክሽን ከተቀየረ በኋላ የትኛው መግለጫ ነው ትክክል? መልስህን አስረዳ።

ג. በጥያቄው መጀመሪያ ላይ የተሰጠውን ችግር የዓላማ ፈንክሽን ብቻ ወደ

$\min\{z = -5x_1 + x_2\}$ ይቀይሩታል።

የዓላማው ፈንክሽን ከተቀየረ በኋላ የትኛው መግለጫ ነው ትክክል? መልስህን አስረዳ።

ד. በጥያቄው መጀመሪያ ላይ የተሰጠውን ችግር የዓላማ ፈንክሽን ብቻ ወደ

$\max\{z = -2x_1 + 2x_2\}$ ይቀይሩታል።

የዓላማው ፈንክሽን ከተቀየረ በኋላ የትኛው መግለጫ ነው ትክክል? መልስህን አስረዳ።

ה. በጥያቄው መጀመሪያ ላይ ለተሰጠው ችግር ተጨማሪ ግዴታ (אילוץ) $2x_1 - x_2 \leq 2$ን ጨመሩለት እና $(-1) < a < 2$ በሚሆንበት በጥያቄው መጀመሪያ ላይ የተሰጠውን ችግር የዓላማ ፈንክሽን ወደ $\max\{z = ax_1 + x_2\}$ ይቀይሩታል። የዓላማው ፈንክሽን የሚቀበለው ሊሆን በሚችለው ክልል ማክሲማል ዋጋ 4 ነው። የa ን ዋጋ አስላ።

/ገጽ 14 ላይ ይቀጥላል/

10. **በዚህ ጥያቄ በመካከላቸው ግንኙነት የሌለ ሁለት ንዑሳን ጥያቄዎች** א-ב **አሉ። *ሁለቱንም* መመለስ አለብህ።**

א. **በፊትህ በሚገኘው የቅርበት መዝገብ** (רשימת הסמיכויות) **የሚወክለው** $G = (V , E)$ **ጠቋሚ ያልሆነ ግራፍ ነው።**

a	→ b → c → d → ‖
b	→ a → c → ‖
c	→ a → b → ‖
d	→ a → f → e → ‖
e	→ d → f → ‖
f	→ e → d → ‖

(1) **በቅርበቱ መዝገብ/ዝርዝር የተወከለውን ግራፍ** G **ን ንደፍ።**

(2) **የተሰጠው ግራፍ** G **የሚያያዝ** (Connected) **ግራፍ ነውን?** **አስረዳ።**

(3) **በተሰጠው ግራፍ ላይ በዋና ነጥብ** a **ጀምርና** <u>**ወደጥልቅ**</u> (DFS) **መፈለግ የሚችል ፈታሽ አልጎሪትም** (אלגוריתם סריקה) **አሠራ። የሚገኘውን ተስፋፊ ዛፍ** (DFS) **ብቻ በደብተርህ ላይ ንደፍ።**
በቅርበት መዝገብ/ዝርዝር አማካኝነት በተሰጠው ወኪሉ ላይ ተመርኮዝ።

(4) **በተሰጠው ግራፍ ላይ በዋና ነጥብ** a **ጀምርና** <u>**ወደጎን**</u> (BFS) **መፈለግ የሚችል ፈታሽ አልጎሪትም** (אלגוריתם סריקה) **አሠራ። የሚገኘውን ተስፋፊ ዛፍ ብቻ** (BFS) **በደብተርህ ላይ ንደፍ።**
በቅርበት መዝገብ/ዝርዝር አማካኝነት በተሰጠው ወኪሉ ላይ ተመርኮዝ።

/ገጽ 15 ላይ ይቀጥላል/

ኃ. (ከንዑስ ጥያቄ א ጋር ግንኙነት የለውም)።

(1) **የመጓጓዣ ችግር/ፕሮብሌም ከፊል መሠረታዊ መፍትሔ መሆን የሚችል እና የ** $u_1, u_2, u_3, v_1, v_2, v_3$ **ዋጋዎች መረጃ በፊትህ ባለው ሠንጠረዥ ውስጥ ተሰጥተዋል።**

መነሻዎች	**መድረሻዎች** 1	2	3	**አቅርቦት (היצע)**	u_i
1	[2] (20)	[5]	[7]	20	2
2	[1]	[1] (10)	[4]	10	0
3	[0]	[1]	[8] (10)	15	0
ፍላጎት (ביקוש)	20	15	10		
v_j	0	1	8		

ሠንጠረዡን ወደደብተርህ ገልብጥ እና <u>**መሠረታዊ**</u> **መፍትሔ መሆን የሚችለው ይገኝ ዘንድ የጎደሉትን የ** u_i**-ዎች ዋጋዎች እና የ**v_j**-ዎች ዋጋዎች እሳቤ ውስጥ በማስገባት አሟላው።**

(**ልብ በል፦ የንዑስ ጥያቄው ቀጣይ ክፍል በሚከተለው ገጽ ላይ ነው።**)

/ገጽ 16 ላይ ይቀጥላል/

(2) የመጓጓዣ ችግር/ፕሮብሌም መሠረታዊ መፍትሔ መሆን የሚችል እና የ u_1, u_2, u_3, v_1, v_2, v_3 ዋጋዎች መረጃ በፊትህ ባለው ሠንጠረዥ ውስጥ ተሰጥተዋል።

መነሻዎች	መድረሻዎች			አቅርቦት	u_i
	1	2	3		
1	[14]	[15] (130)	[17] (50)	180	0
2	[10] (80)	[8] (20)	[14]	100	−7
3	[15]	[20]	[18] (80)	80	1
ፍላጎት	80	150	130		
v_j	17	15	17		

መፍትሔው ኦፕቲማል ነውን? መልስህን አስረዳ።

(3) የመጓጓዣ ችግር/ፕሮብሌም መሠረታዊ መፍትሔ መሆን የሚችል በፊትህ ባለው ሠንጠረዥ ውስጥ ይገኛል፣ እናም $v_1 = 0$ ተሰጥቷል።

መነሻዎች	መድረሻዎች			አቅርቦት	u_i
	1	2	3		
1	[10] (20)	[25]	[30]	20	
2	[10] (30)	[22]	[14] (20)	50	
3	[18]	[20] (40)	[20] (20)	60	
ፍላጎት	50	40	40		
v_j	0				

ሠንጠረዡን ወደደብተርህ ገልብጥ እና የጎደሉትን የ u_1 , u_2 , u_3 , v_2 , v_3 ዋጋዎች አሟላበት።

/ገጽ 17 ላይ ይቀጥላል/

የስሌት ሞዴሎች

በዚህ መንገድ የተማርክ ከሆነ ከ11-12 ካሉት ጥያቄዎች ውስጥ አንዱን መልስ (25 ነጥብ)።

11. በዚህ ጥያቄ በመካከላቸው ግንኙነት የሌለ ሁለት ንዑሳን ጥያቄዎች א-ב አሉ። ሁለቱንም መልስ።

א. በ Java እና በ C# የተጻፈ ስልት ተሰጥቷል።

Java

```java
boolean foo(String str) {
    int cntrA = 0;
    int cntrC = 0;
    for (int i= 0; i < str.length; i++) {
        if (str[i] == 'a') cntrA++;
        if (str[i] == 'c') cntrC++;
    }
    if ((cntrA % 2 == 0) &&
        (cntrC % 3 == 0))
        return true;
    return false;
}
```

C#

```csharp
bool Foo(string str) {
    int cntrA = 0;
    int cntrC = 0;
    for (int i= 0; i < str.Length; i++) {
        if (str[i] == 'a') cntrA++;
        if (str[i] == 'c') cntrC++;
    }
    if ((cntrA % 2 == 0) &&
        (cntrC % 3 == 0))
        return true;
    return false;
}
```

(1) የተሰጠው ስልት true የሚመልስላቸው ቃላት ስብስብ የሆነውን ቋንቋ L ን ከአሌፍ ቤት {a , c} በላይ ጻፍ።

(2) ቋንቋ L ን የሚቀበል ዲተርሚኒስቲክ ውሱን (סופי דטרמיניסטי) አውቶማተን ገንባ።

ב. (ከንዑስ ጥያቄ א ጋር ግንኙነት የለውም።)
ከክትትሎች bbb , aaba , ababa ቢያንስ የአንዱን አንድ ትዕይንት የያዙትን ቃላት በሙሉ የሚቀበል ከአሌፍ ቤት በላይ {a , b} ዲተርሚኒስቲክ ያልሆነ ውስን አውቶማተን ገንባ።

12. ቋንቋ L ከአሌፍ ቤት በላይ {a , b, c} በፊትህ ይገኛል።

$L = \{a^n b^m c^{n+m} \mid n , m > 0\}$

א. ቋንቋ L ን የሚቀበል ካርታ አውቶማተን ገንባ።

ב. ቋንቋ L ን የሚቀበል ቱሪንግ ማሽን (מכונת טיורינג) ገንባ።

/ገጽ 18 ላይ ይቀጥላል/

በአካሎች የሚመራ ፕሮግራሚንግ

በዚህ መንገድ የተማርክ ከሆነና በJava የምትጽፍ ከሆነ ከ13-14 ካሉት ጥያቄዎች ውስጥ አንዱን መልስ (25 **ነጥብ**)።

13. **ኢንተርፌሶች** IOne , ITwo , IThree **እና** IFour **እንደዚሁም ክፍሎች** Alpha , Beta, Gamma **እና** Program **በፊትህ ይገኛሉ።**

```
public interface IOne {
  public boolean firstA(Object x);
  public void firstB(int num);
}

public interface ITwo {
  public int second();
}

public interface IThree {
  public int third();
}
public  interface IFour {
  public int fourth();
}

public class Alpha implements IOne {}
public class Beta implements ITwo, IFour {}
public class Gamma implements ITwo, IThree {}
public class Program  {
  public static void main(String[] args) {}
}
```

ለ. **በእያንዳንዳቸው** Alpha, Beta **እና** Gamma, **ክፍሎች መተግበር የሚገባቸውን ስልቶች ስሞች ጻፍና ለምን እንደሆነ አብራራ።**

/ገጽ 19 ላይ ይቀጥላል/

በንዑስ ጥያቄ አ ላይ የጻፍካቸው ስልቶች በእያንዳንዳቸው ክፍሎች ተተግብረዋል ብለህ ገምት።

ב. ውሳኔ (הגדרה) public class Omega implements IFour extends Beta { }
ትክክለኛ ነውን? ትክክለኛ ካልሆነ — ለምን እንዳልሆነ አብራራ።

ג. በፊትህ ከi - iv ላሉት ክፍሎች ለእያንዳንዳቸው በስልት main ብንጽፈው የሚገኘው ስልት main ትክክለኛ መሆኑን ወይም ትክክለኛ አለመሆኑን ወስን። ውሳኔህን አስረዳ።

i ITwo x1 = new ITwo();

ii Beta b = new Beta();

iii Alpha a = new Alpha();
IOne x2 = a;

iv Gamma c = new Gamma ();
IOne x3 = c ;

ד. በፊትህ ከi - ii ላሉት ፍላጎቶች ለእያንዳንዳቸው ትዕዛዝን በመጻፍ ወይም በስልት main ትዕዛዛት አማካኝነት ሊፈጸሙ መቻላቸውን ወስን። ከተቻለ ተስማሚውን ትዕዛዝ ወይም ተስማሚዎቹን ትዕዛዛት ጻፍ። ካልተቻለ — ለምን እንዳልሆነ አብራራ።

i ስልት second ን በ Beta ዓይነት አካል ላይ ማሠራት።

ii Alpha ዓይነት አካል ይሆን ዘንድ Gamma ዓይነት አካልን መቀየር።

/ገጽ 20 ላይ ይቀጥላል/

14. በዚህ ጥያቄ አንዱ በሌላው የማይወሰኑ <u>አራት</u> ንውሳን ጥያቄዎች አ-ፐ አሉ። <u>ሁሉንም</u> መመለስ አለብህ።

አ. ክፍል A እና ክፍል A ን የሚወርሰው ክፍል B ተሰጥተዋል።

ትዕዛዝ A a1 = new B(); ኮምፒሌሽን (הידור) ይካሄድበትና በትክክለኛ ሁኔታ ይሮጣል።

በፊትህ ከi - iii ላሉት ዐረፍተ ነገሮች ለእያንዳንዳቸው፣ ዐረፍተ ነገሩ ትክክል መሆኑን ወይም አለመሆኑን ወስን። ውሳኔህን አብራራ።

i ቀጥተኛ ለውጥ (casting) ስለሚጠይቅ ትዕዛዝ Object obj = a1; ን ለመጻፍ አይቻልም።

ii ቀጥተኛ ለውጥ (casting) ስለሚጠይቅ ትዕዛዝ A a2 = a1; ን ለመጻፍ አይቻልም።

iii ቀጥተኛ ለውጥ (casting) ስለሚጠይቅ ትዕዛዝ B b1 = a1; ን ለመጻፍ አይቻልም።

በ. በክፍል Program ስልት main ተሰጥቷል፦

```
public static void  main(String[] args) {
  C c = new A();
  B bl = (B) (new A());
  B  b2 = new D();
  A  a = new D();
}
```

በፊትህ ከi - v ካሉት አማራጮች ውስጥ ÷ በተሰጠው ስልት main መሠረት ÷ የትኛው በክፍሎች መካከል ያለ ግንኙነትን እንደሚያሳይ ወስን።

i C ከA ትወርሳለች፣ B ከA ትወርሳለች፣ D ከ A ትወርሳለች.

ii A ከ C ትወርሳለች፣ B ከ A ትወርሳለች፣ D ከ A ትወርሳለች.

iii C ከ A ትወርሳለች፣ B ከ A ትወርሳለች፣ D ከ B ትወርሳለች.

iv A ከ C ትወርሳለች፣ C ከ B ትወርሳለች፣ D ከ A ትወርሳለች.

v A ከC፣ ትወርሳለች፣ B ከA ትወርሳለች፣ D ከB ትወርሳለች፣ C ከ B ትወርሳለች።

/ገጽ 20 ላይ ይቀጥላል/

ጎ. የክፍሎች School እና City ውሳኔዎች (הגדרות) ተሰጥተዋል፦

```
public class City {____(1)____int students; }
public class School extends City {
  void poll() { System.out.println("There are "+ students + " in the school ") ; }
}
```

የኮምፒሌሽን (הידור) ስህተት እንዳይኖር (1) ምልክት የተደረገበትን መሥመር በፊትህ ካሉት አማራጮች ውስጥ በአንዱ አማካኝነት ማሟላት ይቻላል። ተስማሚውን አማራጭ ምረጥ እና ወደደብተርህ ገልብጠው።

- public , private ወይም protected ን መጻፍ።
- public ወይም private ን ብቻ መጻፍ።
- public ወይም protected ን ብቻ መጻፍ።
- protected ን ብቻ መጻፍ።
- public ን ብቻ መጻፍ።
- private ን ብቻ መጻፍ።

ፕ. የክፍል Quest ውሳኔ (הגדרה) በፊትህ ተሰጥቷል፦

```
public class Quest {
    private int  num;
    private static int count = 0;

    public Quest()
    {
        count++;
        num = count;
    }
    public void printNow()
    {
        System.out.println (num + "" + count);
    }
}
```

ከክፍል Quest ስንት አካላትን (ኦብጀክቶችን) ማምረት እንደሚቻል እና ውጽአቱ/አውትፑቱ 35 ይሆን ዘንድ በየትኛው ላይ ስልት printNowን ማሠራት እንደሚቻል ወስን።

/ገጽ 22 ላይ ይቀጥላል/

በአካሎች የሚመራ ፕሮግራሚንግ

በዚህ መንገድ የተማርክ ከሆነና በC# የምትጽፍ ከሆነ ከ15-16 ካሉት ጥያቄዎች ውስጥ አንዱን መልስ (25 **ነጥብ)።**

15. **ኢንተርፌሶች** IOne , ITwo , IThree **እና** IFour **እንደዚሁም ክፍሎች** Alpha , Beta, Gamma **እና** Program **በፊትህ ይገኛሉ።**

```
interface IOne {
  bool FirstA(Object x);
  void FirstB(int num);
}

interface ITwo {
  int Second();
}

interface IThree {
  int Third();
}
interface IFour {
  int Fourth();
}

public class Alpha : IOne {}
public class Beta : ITwo, IFour {}
public class Gamma : ITwo, IThree {}
public class Program {
  public static void Main() {}
}
```

ሀ. **በእያንዳንዳቸው** Alpha, Beta **እና** Gamma, **ክፍሎች መተግበር የሚገባቸውን ስልቶች ስሞች ጻፍና ለምን እንደሆነ አብራራ።**

/ገጽ 23 ላይ ይቀጥላል/

በንዑስ ጥያቄ ሀ ላይ የጻፍካቸው ስልቶች በእያንዳንዳቸው ክፍሎች ተተግብረዋል ብለህ ገምት።

ኀ. ውሳኔ (הגדרה) public class Omega : IFour, Beta { } ትክክለኛ ነውን? ትክክለኛ ካልሆነ — ለምን እንዳልሆነ አብራራ።

ኁ. በፊትህ ከi - iv ላሉት ክፍሎች ለእያንዳንዳቸው በMain ብንጽፈው የሚገኘው ስልት Main ትክክለኛ መሆኑን ወይም ትክክለኛ አለመሆኑን ወስን። ውሳኔህን አስረዳ።

i ITwo x1 = new ITwo();

ii Beta b = new Beta();

iii Alpha a = new Alpha();

IOne x2 = a;

iv Gamma c = new Gamma ();
IOne x3 = c ;

ፕ. በፊትህ ከi - ii ላሉት ፍላጎቶች ለእያንዳንዳቸው ትዕዛዝን በመጻፍ ወይም በMain ትዕዛዛት አማካኝነት ሊፈጸሙ መቻላቸውን ወስን። ከተቻለ — ተስማሚውን ትዕዛዝ ወይም ተስማሚዎቹን ትዕዛዛት ጻፍ። ካልተቻለ — ለምን እንዳልሆነ አብራራ።

i ስልት Second ን በ Beta ዓይነት አካል ላይ ማሠራት።

ii Alpha ዓይነት አካል ይሆን ዘንድ Gamma ዓይነት አካልን መቀየር።

/ገጽ 24 ላይ ይቀጥላል/

16. በዚህ ጥያቄ አንዱ በሌላው የማይወሰኑ አራት ንዑሳን ጥያቄዎች א-ד **አሉ። ሁሉንም መመለስ አለብህ።**

א. **ክፍል** A **እና ክፍል** A **ን የሚወርሰው ክፍል** B **ተሰጥተዋል።**

ትዕዛዝ A a1 = new B(); **ኮምፒሌሽን** (הידור) **ይካሄድበትና በትክክለኛ ሁኔታ ይሮጣል።**

በፊትህ ከi - iii **ላሉት ዐረፍተ ነገሮች ለእያንዳንዳቸው፣ ዐረፍተ ነገሩ ትክክል መሆኑን ወይም አለመሆኑን ወስን። ውሳኔህን አብራራ።**

i **ቀጥተኛ ለውጥ** (casting) **ስለሚጠይቅ ትዕዛዝ** Object obj = a1; **ን ለመጻፍ አይቻልም።**

ii **ቀጥተኛ ለውጥ** (casting) **ስለሚጠይቅ ትዕዛዝ** A a2 = a1; **ን ለመጻፍ አይቻልም።**

iii **ቀጥተኛ ለውጥ** (casting) **ስለሚጠይቅ ትዕዛዝ** B b1 = a1; **ን ለመጻፍ አይቻልም።**

ב. **በክፍል** Program **ስልት** Main **ተሰጥቷል፦**

```
public static void  Main(string[] args) {
  C c = new A();
  B bl = (B) (new A());
  B  b2 = new D();
  A  a = new D();
}
```

ከi - v **ካሉት አማራጮች ውስጥ ÷ በተሰጠው ስልት** Main **መሠረት ÷ የትኛው በክፍሎች መካከል ያለ ግንኙነትን እንደሚያሳይ ወስን።**

i C ከA **ትወርሳለች፣** B ከA **ትወርሳለች፣** D ከ A **ትወርሳለች።**

ii A ከ C **ትወርሳለች፣** B ከ A **ትወርሳለች፣** D ከ A **ትወርሳለች።**

iii C ከ A **ትወርሳለች፣** B ከ A **ትወርሳለች፣** D ከ B **ትወርሳለች።**

iv A ከ C **ትወርሳለች፣** C ከ B **ትወርሳለች፣** D ከ A **ትወርሳለች።**

v A ከ C **ትወርሳለች፣** B ከA **ትወርሳለች፣** D ከ B **ትወርሳለች፣** C ከ B **ትወርሳለች።**

/ገጽ 25 ላይ ይቀጥላል/

ጎ. የክፍሎች School **እና** City **ውሳኔዎች** (הגדרות) **ተሰጥተዋል፦**

```
public class City{____(1)_____int students;  }
public  class  School : City {
  void Poll() { Console.WriteLine("There are "+ students + " in the school ") ; }
}
```

የኮምፒሌሽን (הידור) **ስሀተት እንዳይኖር** (1) **ምልክት የተደረገበትን መሥመር በፊትህ ካሉት አማራጮች ውስጥ በአንዱ አማካኝነት ማሟላት ይቻላል። ተስማሚውን አማራጭ ምረጥ እና ወደደብተርህ ገልብጠው።**

- public, private **ወይም** protected **ን መጻፍ።**
- public **ወይም** private **ን ብቻ መጻፍ።**
- public **ወይም** protected **ን ብቻ መጻፍ።**
- protected **ን ብቻ መጻፍ።**
- public **ን ብቻ መጻፍ።**
- private **ን ብቻ መጻፍ።**

ገ. የክፍል Quest **ውሳኔ** (הגדרה) **በፊትህ ተሰጧል፦**

```
public class Quest  {
      private int num;
      private static int count = 0;

      public Quest()
      {
            count++;
            num = count;
      }
      public void PrintNow()
      {
            Console.WriteLine(num+""+count);
      }
}
```

ከክፍል Quest **ስንት አካላትን (ኦብጀክቶችን) ማምረት እንደሚቻል እና ውጽአቱ/ አውትፑቱ** 35 **ይሆን ዘንድ በየትኛው ላይ ስልት** PrintNow**ን ማሠራት እንደሚቻል ወስን።**

መልካም ዕድል!

בהצלחה!

መብቱ የእስራኤል መንግሥት ነው።
መገልበጥም ሆነ ማባዛት እንዲሁም ማሳተም በት/ሚኒስቴር ፈቃድ ብቻ ነው።

זכות היוצרים שמורה למדינת ישראל
אין להעתיק או לפרסם אלא ברשות משרד החינוך