

Тип экзамена:

סוג הבחינה:

а. на аттестат зрелости для средних школ

א. בגרות לבתי ספר על-יסודיים

б. на аттестат зрелости для экстернов

ב. בגרות לנבחנים חיצוניים

Время проведения экзамена: лето 2016 года

מועד הבחינה: קיץ תשע"ו, 2016

Номер вопросника: 899205, 603

מספר השאלון: 899205, 603

Перевод на русский язык (5)

תרגום לרוסית (5)

Компьютеры "бет"

2 единицы обучения (дополнение до 5-ти единиц)

מדעי המחשב ב'

2 יחידות לימוד (השלמה ל-5 יח"ל)

Указания экзаменуемым

а. Продолжительность экзамена: три часа.

א. משך הבחינה: שלוש שעות.

б. Строение вопросника и ключ к оценке:

ב. מבנה השאלון ומפתח ההערכה:

в этом вопроснике два раздела.

בשאלון זה שני פרקים.

Раздел первый: в этом разделе четыре вопроса.

פרק ראשון — בפרק זה ארבע שאלות,

Вы должны ответить на два из них.

ומהן יש לענות על שתיים.

$(2 \times 25) = 50$ баллов

פרק שני — בפרק זה שאלות בארבעה

Раздел второй: в этом разделе вопросы из четырех

מסלולים שונים, ענה על שאלות רק

различных курсов. Вы должны ответить только

במסלול שלמדת, לפי ההוראות בקבות

на вопросы курса, который вы изучали, согласно

השאלות במסלול זה.

указаниям этого курса. $(2 \times 25) = 50$ баллов

50 נקודות

Всего — 100 баллов

25x2 — 50 נקודות

סה"כ — 100 נקודות

в. Разрешенный вспомогательный материал:

ג. חומר עזר מותר בשימוש:

любой вспомогательный материал, за исключением
компьютера с возможностью программирования.

כל חומר עזר, חוץ ממחשב הניתן לתכנות.

г. Особые указания:

ד. הוראות מיוחדות:

1. Все программы, которые вы должны написать

1. את כל התכניות שאתה נדרש לכתוב

на языке программирования в первом разделе,

בשפת מחשב בפרק הראשון כתוב

следует писать только на одном языке — Java или C# .

בשפה אחת בלבד —

Java או C# .

2. На внешней обложке тетради укажите язык

2. רשום על הכריכה החיצונית של

программирования, который вы используете:

המחברת באיזו שפה אתה כותב —

Java или C# .

Java או C# .

3. На внешней обложке тетради укажите название

3. רשום על הכריכה החיצונית של

курса, который вы изучали, одного из четырех

המחברת את שם המסלול שלמדת,

следующих курсов: “Компьютерные системы и

המסלול הוא אחד מארבעת המסלולים

асемблер” [מערכות מחשב ואסמבלי],

האלה: מערכות מחשב ואסמבלי,

“Введение в исследование операций” [מבוא לחקר ביצועים],

מבוא לחקר ביצועים, מודלים

“Вычислительные модели” [מודלים חישוביים],

חישוביים, תכנות מונחה עצמים.

“Объектно-ориентированное программирование”

הערה: בתכניות שאתה כותב לא יורדו לך

[תכנות מונחה עצמים].

נקודות, אם תכתוב אות גדולה

Примечание: в программах, которые вы должны

במקום אות קטנה או להפך.

написать, за использование большой буквы вместо

маленькой или наоборот баллы сниматься не будут.

כתוב במחברת הבחינה בלבד, בעמודים נפרדים, כל מה שברצונך לכתוב בטיטות (ראשי פרקים, חישובים וכדומה). רשום “טיטות” בראש כל עמוד טיטות.

רשום טיטות לשהן על דפים מחוץ למחברת הבחינה עלול לגרום לפסילת הבחינה!

Все черновые записи (тезисы, вычисления и т.п.) делайте только на отдельных страницах экзаменационной тетради. Напишите слово «черновик» в начале каждой страницы, отведенной вами под черновик. Выполнение любых черновых записей на листах, не относящихся к экзаменационной тетради, может привести к тому, что экзамен будет аннулирован!

Желаем успеха!

בהצלחה!

Вопросы

В данном вопроснике два раздела: **раздел первый** и **раздел второй**.
Вы должны ответить на вопросы обоих разделов, согласно инструкциям каждого раздела.

РАЗДЕЛ ПЕРВЫЙ (50 баллов)

Внимание: во всех вопросах, где требуется реализация [מימוש], разрешается пользоваться методами классов Список, Очередь, Стек, Двоичное дерево и Звено [רשימה, תור, מחסנית, עץ בינרי וחוליה], не реализуя их. Если вы используете дополнительные методы, вы должны их реализовать [לממש אותם].

Ответьте на два из вопросов 1-4 (за каждый вопрос – 25 баллов).

1. В этом вопросе два пункта א-ב, между которыми нет связи.

Ответьте на вопросы обоих пунктов.

(א) Перед вами метод, получающий параметр q – очередь [תור] целых чисел, отсортированных по возрастанию. Все числа в очереди отличны друг от друга.

Считайте, что очередь содержит как минимум два целых числа.

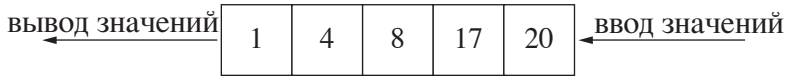
Java

```
public static int what(Queue<Integer> q)
{
    int x = q.remove();
    int y = q.head();
    q.insert(x);
    if (x > y)
        return 0;
    return (what(q) + (y - x));
}
```

C#

```
public static int What(Queue<int> q)
{
    int x = q.Remove();
    int y = q.Head();
    q.Insert(x);
    if (x > y)
        return 0;
    return (What(q) + (y - x));
}
```

(1) Дана очередь q1 .



Проследите за выполнением метода what на языке Java
и метода What на языке C# для данной очереди q1
и запишите возвращаемое значение.

Покажите рекурсивную трассировку [רקורסיה].

Трассировка должна отображать значения x и y , а также
состояние очереди после каждого рекурсивного вызова.

- (2) (i) Что возвращает метод what на языке Java и метод What
на языке C# для некоторой очереди q , содержащей целые
числа, отсортированные по возрастанию и отличные друг
от друга?
- (ii) Как выглядит очередь после выполнения данного метода?

(3) Если условие $if(x > y)$ заменить на условие

`if(q.isEmpty())` на языке Java

или

`if(q.IsEmpty())` на языке C# ,

произойдет ошибка выполнения [שגיאת זמן ריצה].

Какая ошибка произойдет? Объясните свой ответ.

Обратите внимание: пункт (2) этого вопроса на следующей странице.

(ב) Нет связи с пунктом (א).

Перед вами метод, получающий одномерный массив [מערך חד-ממדי] `ar` целого типа и два целых числа `x` и `y`. Эти числа больше, чем 0, или равны 0 и меньше, чем длина данного массива, причем $x \leq y$.

Java	C#
<pre>public static int go(int[] ar , int x , int y) { if (x == y) return (ar[x] % 10) ; if ((ar[x] % 10) < (ar[y] % 10)) return go(ar, x + 1, y); return go(ar, x, y-1); }</pre>	<pre>public static int Go(int[] ar , int x , int y) { if (x == y) return (ar[x] % 10) ; if ((ar[x] % 10) < (ar[y] % 10)) return Go(ar, x + 1, y); return Go(ar, x, y-1); }</pre>

Перед вами массив `ar1` целого типа.

	0	1	2	3	4	5	6
ar1	21	219	166	61	127	60	34

(1) Что вернет вызов `go(ar1 , 2 , 5)` на Java или `Go(ar1 , 2 , 5)` на C# ?

Покажите рекурсивную трассировку [דָּרֻמ]. В трассировке укажите значения:

`x` , `y` , `ar1[x]`, `ar1[y]` .

(2) Что вернет вызов `go(ar1, 0, ar1.length-1)` на языке Java или `Go(ar1, 0, ar1.Length-1)` на языке C# ?

(3) Напишите, что выполняет метод

`public static int go(int[] ar , int x , int y)` на языке Java

`public static int Go(int[] ar , int x , int y)` на языке C#

для некоторого массива `ar` целого типа и двух целых чисел `x` и `y` , больших, чем 0, или равных 0, и меньших, чем длина массива, таких, что $x \leq y$.

/продолжение на странице 5/

2. **Дерево чисел** – это непустое двоичное дерево [עץ בינארי] целого типа, значения вершин которого это целые числа большие, чем 0, и отличные друг от друга.

Для **дерева чисел** определен метод "Маршрут по возрастанию". Метод возвращает true, если в дереве существует такой маршрут с началом в корне дерева и концом в одном из его листьев, в котором значения вершин отсортированы по возрастанию от корня к листу.

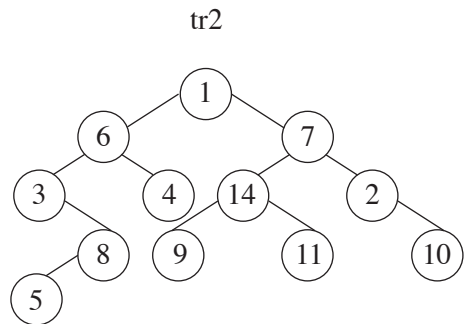
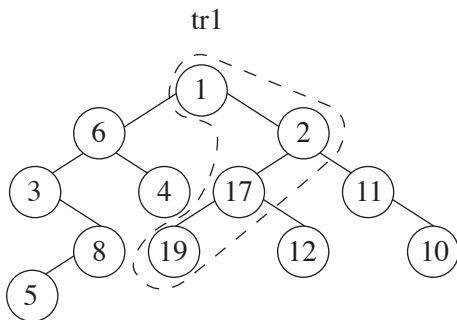
Если такого маршрута в дереве нет, метод возвращает false.

Например:

Для **дерева чисел** tr1 метод "Маршрут по возрастанию" вернет true .

Маршрут обозначен пунктирной линией.

Для **дерева чисел** tr2 метод "Маршрут по возрастанию" вернет false .



Реализуйте на языке Java или C# метод "Маршрут по возрастанию"

для **дерева чисел** tr .

Заголовок метода на Java: `public static boolean upPath(BinNode<Integer> tr)`

Заголовок метода на C#: `public static bool UpPath(BinNode<int> tr)`

3. Перед вами класс Кольцо – **Ring** с двумя атрибутами [תכונות]:
- размер кольца типа строка ("S" – маленькое кольцо, "L" – большое кольцо);
 - целое число, обозначающее цвет кольца.

Java

```
public class Ring
{
    private String size; //размер кольца
    private int color; //цвет кольца
    public Ring()
    {
        this.size = "L";
        this.color = 0;
    }
    public Ring(String str, int c)
    {
        this.size = str;
        this.color = c;
    }
    public String getSize()
    {
        return this.size;
    }
    public int getColor()
    {
        return this.color;
    }
}
```

C#

```
public class Ring
{
    private string size; //размер кольца
    private int color; //цвет кольца
    public Ring()
    {
        this.size = "L";
        this.color = 0;
    }
    public Ring(string str, int c)
    {
        this.size = str;
        this.color = c;
    }
    public string GetSize()
    {
        return this.size;
    }
    public int GetColor()
    {
        return this.color;
    }
}
```

Перед вами интерфейс класса Шест – **Pole**.

Java	C#	Метод
public Pole()	public Pole()	Метод, создающий пустой шест. Временная сложность [סיבוכיות זמן הריצה] метода $O(1)$
public void add(Ring r)	public void Add(Ring r)	Метод, надевающий кольцо r на вершину шеста. Временная сложность метода $O(1)$
public Ring remove()	public Ring Remove()	Метод, возвращающий верхнее кольцо шеста. Метод также убирает верхнее кольцо с шеста. Временная сложность метода $O(1)$
public boolean isEmpty()	public bool IsEmpty()	Если шест пустой, метод возвращает true. Иначе - метод возвращает false. Временная сложность метода $O(1)$
public void sort()	public void Sort()	Метод, сортирующий кольца на шесте так: большие кольца "располагаются" на шесте внизу, а маленькие кольца над ними.

(א) Реализуйте [ממש] в классе **Pole** метод sort() на языке Java или метод Sort() на языке C# .

Можно использовать остальные методы класса **Pole**, не реализуя их.
В вашем ответе можно использовать только методы классов **Pole** и **Ring**.

(ב) Какова временная сложность [סיבוכיות זמן הריצה] метода, реализованного вами в пункте (א)? Обоснуйте ваш ответ.

/продолжение на странице 8/

4. **Обратите внимание:** у этого вопроса два варианта:
один вариант на языке Java на стр. 8-9
и другой вариант на языке C# на стр. 10-11.
Выберите вариант языка, который вы изучали.

Для решающих на языке Java:

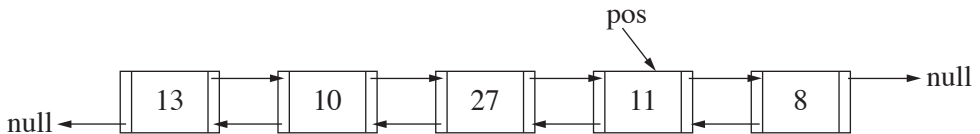
Определим **двунаправленный список** как упорядоченный набор [אוסף סדור] звеньев типа `BinNode<Integer>`, связанных следующим образом:

для каждой пары звеньев `p1`, `p2` в списке, если соблюдается условие `p1.getRight() == p2`, то соблюдается также `p2.getLeft() == p1`.

В **двунаправленном списке** есть как минимум два звена.

Иными словами: каждое звено списка – кроме правого крайнего и левого крайнего звена – указывает на предыдущее звено и на последующее звено.

Перед вами пример **двунаправленного списка** и переменной `pos` типа `BinNode<Integer>`, указывающей на некоторое звено в двунаправленном списке.



Метод `firstLeft` получает указатель `pos` типа `BinNode<Integer>`, не равный `null`, указывающий на некоторое звено в двунаправленном списке. Метод возвращает крайнее левое звено в данном списке.

Метод `firstRight` получает указатель `pos` типа `BinNode<Integer>`, не равный `null`, указывающий на некоторое звено в двунаправленном списке.

Метод возвращает крайнее правое звено в данном списке.

(x) Перед вами скелет метода `firstLeft`.

Скопируйте его в вашу тетрадь и дополните так, чтобы метод выполнял требуемое действие.

```
public static BinNode<Integer> firstLeft(BinNode<Integer> pos)
```

```
{
    while ( _____ )
        pos = _____ ;
    return _____ ;
}
```

/продолжение на странице 9/

- (ב) Перед вами метод `what(BinNode<Integer> pos)`, получающий указатель на некоторое звено в **двунаправленном списке** и возвращающий булево значение [ערך בוליאני]. **Двунаправленный список** содержит как минимум 3 звена.

- (1) Проследите за выполнением данного метода для переменной `pos` и двунаправленного списка, приведенных в начале вопроса.

Трассировка [מעקב] должна отображать двунаправленный список и значения переменных `pos`, `left`, `right`, `sum`.

```
public static boolean what(BinNode<Integer> pos)
{
    BinNode<Integer> left = firstLeft(pos);
    BinNode<Integer> right = firstRight(pos);

    int sum = left.getValue() + right.getValue();
    left = left.getRight();
    right = right.getLeft();

    while ((left != right) && (left.getRight() != right) &&
           (left.getValue() + right.getValue() == sum))
    {
        left = left.getRight();
        right = right.getLeft();
    }
    if (left == right)
        return right.getValue() == sum;

    if (left.getRight() == right)
        return left.getValue() + right.getValue() == sum;
    return false;
}
```

- (2) Определите, можно ли заменить три последние строки метода – строки, заключенные в рамку – следующей командой:

```
return left.getValue() + right.getValue() == sum;
```

Обоснуйте ваш ответ.

/продолжение на странице 10/

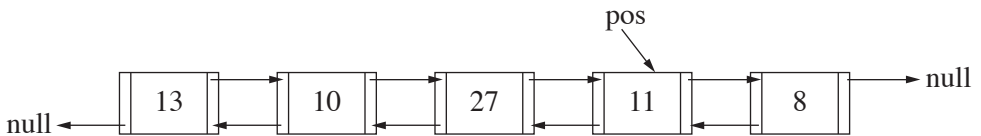
Для решающих на языке C#

Определим **двунаправленный список** как упорядоченный набор [אוסף סדור] звеньев типа `BinNode<int>`, связанных следующим образом: для каждой пары звеньев `p1`, `p2` в списке, если соблюдается условие `p1.GetRight() == p2`, то соблюдается также `p2.GetLeft() == p1`.

В **двунаправленном списке** есть как минимум два звена.

Иными словами: каждое звено списка – кроме правого крайнего и левого крайнего звена – указывает на предыдущее звено и на последующее звено.

Перед вами пример **двунаправленного списка** и переменной `pos` типа `BinNode<int>`, указывающей на некоторое звено в двунаправленном списке.



Метод `FirstLeft` получает указатель `pos` типа `BinNode<int>`, не равный `null`, указывающий на некоторое звено в двунаправленном списке. Метод возвращает крайнее левое звено в данном списке.

Метод `FirstRight` получает указатель `pos` типа `BinNode<int>`, не равный `null`, указывающий на некоторое звено в двунаправленном списке.

Метод возвращает крайнее правое звено в данном списке.

(x) Перед вами скелет метода `FirstLeft`.

Скопируйте его в вашу тетрадь и дополните так, чтобы метод выполнял требуемое действие.

```
public static BinNode<int> FirstLeft(BinNode<int> pos)
```

```
{  
    while ( _____ )  
        pos = _____ ;  
    return _____ ;  
}
```

(ב) Перед вами метод `What(BinNode<int> pos)`, получающий указатель на некоторое звено в **двунаправленном списке** и возвращающий булево значение [ערך בוליאני]. **Двунаправленный список** содержит как минимум 3 звена.

(1) Проследите за выполнением данного метода для переменной `pos` и двунаправленного списка, приведенных в начале вопроса.
Трассировка [מעקב] должна отображать двунаправленный список и значения переменных `pos`, `left`, `right`, `sum`.

```
public static bool What(BinNode<int> pos)
{
    BinNode<int> left = FirstLeft(pos);
    BinNode<int> right = FirstRight(pos);

    int sum = left.GetValue() + right.GetValue();
    left = left.GetRight();
    right = right.GetLeft();

    while ((left != right) && (left.GetRight() != right) &&
           (left.GetValue() + right.GetValue() == sum))
    {
        left = left.GetRight();
        right = right.GetLeft();
    }
    if (left == right)
        return right.GetValue() == sum;

    if (left.GetRight() == right)
        return left.GetValue() + right.GetValue() == sum;
    return false;
}
```

(2) Определите, можно ли заменить три последние строки метода - строки, взятые в рамку – следующей командой:

```
return left.GetValue() + right.GetValue() == sum;
```

Обоснуйте ваш ответ.

/продолжение на странице 12/

РАЗДЕЛ ВТОРОЙ (50 баллов)

В данном разделе вопросы четырёх разных курсов:

- Компьютерные системы и ассемблер [מערכות מחשב ואסמבלי], стр. 12-18.
- Введение в исследование операций [מבוא לחקר ביצועים], стр. 19-27.
- Вычислительные модели [מודלים חישוביים], стр. 28-29.
- Объектно-ориентированное программирование [תכנות מונחה עצמים] на языке Java, стр. 30-39; объектно-ориентированное программирование на языке C#, стр. 40-49.

Ответьте только на вопросы того курса, который вы изучали.

Компьютерные системы и Ассемблер [מערכות מחשב ואסמבלי]

Если вы изучали этот курс, ответьте на два вопроса из вопросов 5-8 (за каждый вопрос – 25 баллов).

5. В данном вопросе два не связанных между собой пункта (א)-(ב). Ответьте на вопросы обоих этих пунктов.

(א) Во фрагменте данных [מקטע הנתונים] определены следующие данные:

VEC1 DB 45H , 26H , 32H , 82H

VEC2 DB 4 DUP(0)

Перед вами фрагмент [קטע] программы на Ассемблере.

```
START:    MOV    CL , 4
          XOR    CH , CH
          XOR    SI , SI
NEXT:     PUSH   CX
          MOV    AL , VEC1[SI]
          MOV    AH , AL
          AND    AL , 0FH
          MOV    CL , 4
          SHR    AH , CL
          MUL    AH    ;(*)
          MOV    VEC2[SI] , AL
          INC    SI
          POP    CX
          LOOP   NEXT
SOF:      NOP
```

/продолжение на странице 13/

- (1) Проследите с помощью таблицы трассировки [טבלת מעקב] за выполнением данного фрагмента и начертите VEC2 по окончании выполнения.

Таблица трассировки должна содержать столбец для каждого из регистров [אוגרים] AL , AH , CL , CH , SI .

Начертите стек [מחסנית] на каждом этапе.

- (2) Что выполняет данный фрагмент программы?
- (3) В данном фрагменте программы вместо команды, обозначенной (*), написали команду: IMUL AH
Изменится ли выполнение данного фрагмента программы?
Обоснуйте ваш ответ.

- (ג) Нет связи с пунктом (א).

Перед вами фрагмент [קטע] программы на Ассемблере.

```
MOV    AX , 14H
MOV    BL , 0AH
DIV     BL
MOV     CL , 4
ROL     AL , CL
OR      AL , AH
```

- (1) Проследите с помощью таблицы трассировки [טבלת מעקב] за выполнением данного фрагмента.

Таблица трассировки должна содержать столбец для каждого из регистров [אוגרים] AX , AL , AH , BL , CL .

- (2) Что выполняет данный фрагмент программы?

6. Чтобы проверить, делится ли число на 6 без остатка, число делят на 2 .
Если оно делится на 2 без остатка, полученный результат делят на 3 .
Если результат делится на 3 без остатка, число делится на 6 без остатка.
- (א) Напишите на Ассемблере процедуру с именем DIVBYTWO, которая получит через стек [מחסנית] число из 16 битов [סיביות] и сохранит 0 в AL , если число без остатка делится на 2 .
Иначе – процедура сохранит 1 в AL .
При написании процедуры нельзя использовать команду DIV .
- (ב) Напишите на Ассемблере процедуру с именем DIVBYTHREE, которая получит через стек число из 16 битов и сохранит 0 в AL , если число без остатка делится на 3 .
Иначе – процедура сохранит 1 в AL .
При написании процедуры можно использовать любые команды.
- (ג) Напишите на Ассемблере процедуру с именем DIVBYSIX, которая получит через стек число из 16 битов и сохранит 0 в AL , если число без остатка делится на 6 .
Иначе – процедура сохранит 1 в AL .
Вы должны использовать процедуры, написанные вами для пунктов (א) и (ב).
- (ד) Во фрагменте данных [מקטע הנתונים] определен массив ARR из 8 элементов, длина каждого из которых – одно слово. Массив содержит числа.
Напишите на Ассемблере фрагмент [קטע] программы, который выведет на экран [ידפוס] количество элементов массива, значения которых делятся на 6 без остатка.
Вы должны использовать процедуру, написанную вами для пункта (ג).

7. В данном вопросе два не связанных между собой пункта (א)-(ב). Ответьте на вопросы обоих этих пунктов.

(א) Напишите на Ассемблере фрагмент [טקט] программы для кодировки символа [הצפנת תנן].

Символ определен во фрагменте данных [מקטע הנתונים] следующим образом:

TAV DB ?

Кодировка выполняется так:

Подсчитывается число включенных битов [סיביות דלוקות] в символе.

Если это число четное, производится круговой сдвиг [הזזה מעגלית]

битов символа вправо столько раз, сколько включенных битов было в символе.

Если число нечетное, производится круговой сдвиг битов символа влево столько раз, сколько включенных битов было в символе.

(ב) Нет связи с пунктом (א).

Для каждого из приведенных ниже утверждений (1)-(6) определите, верно оно или нет. Если утверждение неверно, поясните почему.

(1) Перед вами фрагмент [טקט] программы на Ассемблере.

MOV AX , 8

MOV BX , 2

DIV BX

По окончании выполнения данного фрагмента регистр [אוגר] AX обязательно будет содержать 4.

(2) Перед вами фрагмент программы на Ассемблере.

MOV AL , 56

ADD AL , 200

JZ STOP

INC AL

STOP:

По окончании выполнения данного фрагмента регистр AX будет содержать 1.

Обратите внимание: продолжение пункта вопроса на следующей странице.
/продолжение на странице 16/

(3) Перед вами фрагмент программы на Ассемблере.

```
ARRAY    DW 1, 2, 3, 4
MOV      BX , ARRAY
ADD      BX , 2
MOV      AX , [BX]
```

По окончании выполнения данного фрагмента регистр AX будет содержать 3.

(4) Перед вами фрагмент программы на Ассемблере.

```
MOV      CX , 3
MOV      AX , 1
DO:
SHL      AX , 1
LOOP     DO
```

По окончании выполнения данного фрагмента регистр AX будет содержать 8.

(5) Перед вами фрагмент программы на Ассемблере.

```
MOV      AX , 11000001B
MOV      BX , 01000001B
```

Число, содержащееся в AX , обязательно больше числа, содержащегося в BX.

(6) Перед вами команда:

```
OR      AL , 3
```

После выполнения данной команды значение в AL всегда нечетное.

8. В данном вопросе два не связанных между собой пункта (א)-(ב).

Ответьте на вопросы обоих этих пунктов.

(א) В регистре [אוגר] AL хранится число 2, а в регистре BL хранится число 5. Требуется сохранить в регистре DX сумму чисел от AL до BL (включительно).

Перед вами три фрагмента [קטעים] i-iii на Ассемблере.

Проследите с помощью таблицы трассировки [טבלת מעקב]

за выполнением каждого из данных фрагментов и определите, выполняет ли он требуемое действие.

```
i          MOV    DX , 0
           MOV    AH , 0
           MOV    CL , AL
           SUB    CL , BL
AGAIN:     ADD    DX , AX
           INC    AL
           LOOPE  AGAIN
           NOP
```

```
ii          XOR    DX , DX
           MOV    BH , 0
AGAIN:     ADD    DX , BX
           DEC    BL
           CMP    BL , AL
           JGE    AGAIN
           NOP
```

```
iii         XOR    DX , DX
           XOR    AX , AX
           MOV    BX , 0
AGAIN:     ADD    DX , AX
           ADD    AX , 1
           CMP    AX , BX
           JL     AGAIN
           NOP
```

Обратите внимание: пункт (ב) этого вопроса на следующей странице.

/продолжение на странице 18/

(ב) Нет связи с пунктом (א).

Перед вами фрагмент [קטע] программы на Ассемблере.

Обратите внимание: числа шестнадцатиричные [הקסדצימליים].

```
START:  MOV  AX , 0C72AH
        MOV  BX , 0A98DH
        SHR  AX , 1
        OR   AL , 17H
        NOT  BH
        ADD  AX , BX
```

Проследите с помощью таблицы трассировки [טבלת מעקב]

за выполнением данного фрагмента.

Таблица трассировки должна содержать столбец для каждого из флагов [דגלים] : CF, OF, SF, ZF

и для каждого из регистров [אוגרים] : AX, BX.

Введение в исследование операций [מבוא לחקר ביצועים]

Если вы изучали этот курс, ответьте на два из вопросов 9-12
(за каждый вопрос – 25 баллов).

9. Дана задача линейного программирования:

$$\max \{z = (2 + 2k)x_1 + 2x_2\}$$

при следующих ограничениях:

$$(1) \ 2x_1 + x_2 \leq 10$$

$$(2) \ x_1 + x_2 \leq 6$$

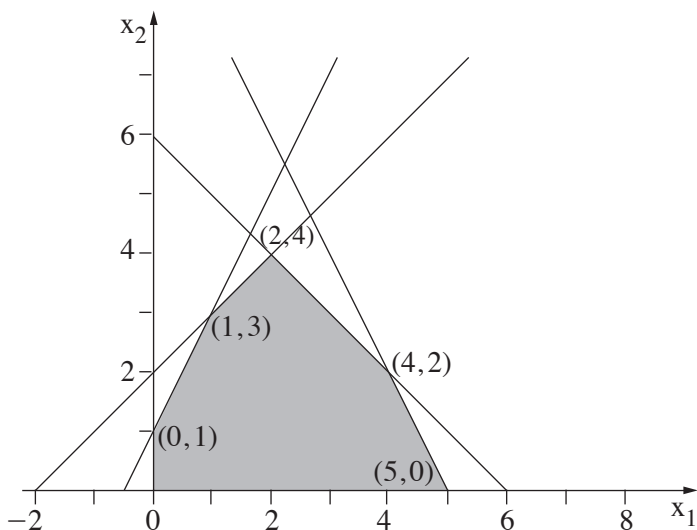
$$(3) \ -x_1 + x_2 \leq 2$$

$$(4) \ -2x_1 + x_2 \leq 1$$

$$x_1 \geq 0$$

$$x_2 \geq 0$$

Перед вами чертеж области возможных решений данной задачи.



Обратите внимание: пункты вопроса на следующей странице.

Все пункты (א)-(ז) относятся к данной задаче линейного программирования.

Пункты не связаны друг с другом. Ответьте на вопросы всех пунктов.

(א) Перед вами два **не связанных друг с другом** подпункта (1)-(2).

В каждом из них дано определенное значение параметра k .

(1) $k = -1$

(2) $k = -3$

Для каждого из подпунктов (1)-(2) даны четыре утверждения i-iv:

i Существует единственное оптимальное решение.

ii Существует бесконечное множество оптимальных решений.

iii Оптимальное решение не ограничено [לא חסום].

iv Оптимального решения нет.

Для каждого из подпунктов только одно из утверждений i-iv верно.

Для каждого из подпунктов (1)-(2) определите, какое из утверждений верно, скопируйте его в вашу тетрадь и обоснуйте свой выбор.

- Если вы считаете, что верно утверждение i, вы должны найти единственное оптимальное решение для этого подпункта, а также значение целевой функции для найденного решения.
- Если вы считаете, что верно утверждение ii, вы должны найти общее оптимальное решение [הפתרון האופטימלי הכללי] для этой задачи, а также значение целевой функции в области оптимальных решений.

(ב) Для каких значений k оптимальное решение задачи линейного программирования, данной в начале вопроса – это $(2, 4)$? Обоснуйте свой ответ.

(ג) В задаче линейного программирования, данной в начале вопроса, отменяют ограничение $x_1 \geq 0$, то есть $-\infty \leq x_1 \leq \infty$.

Существует ли значение k , для которого оптимальное решение будет не ограничено [לא חסום]? Обоснуйте свой ответ.

(ד) В задаче линейного программирования, данной в начале вопроса, вместо ограничения $x_2 \geq 0$ дано ограничение $x_2 \geq 4$.

Существует ли оптимальное решение у задачи после изменения ограничения? Если существует, запишите его. Если нет, объясните, почему оптимального решения нет.

10. В данном вопросе два не связанных между собой пункта, (א) и (ב).

Ответьте на вопросы обоих пунктов.

(א) $G = (V, E)$ – **ориентированный** граф [גרף מכוון], представленный матрицей смежностей [מטריצת סמיכויות]:

	a	b	c	d	e
a	0	1	1	0	0
b	0	0	0	1	0
c	0	1	0	0	0
d	0	0	1	0	1
e	0	1	0	0	0

- (1) Начертите граф G , представленный данной матрицей.
 - (2) Найдите компоненты сильной связности [רכיבי הקשירות החזקה] (Strong Connected Components – [רק"חים]) в данном графе. Для каждого найденного компонента запишите множество его вершин.
 - (3) Найдите в этом графе цикл [מעגל] минимальной длины, содержащий четное число ребер, и начертите его в вашей тетради.
 - (4) Каково минимальное число ребер, которые необходимо добавить в данный граф, чтобы он содержал единственный компонент сильной связности? Какое ребро (или какие ребра) требуется добавить?
- (ב) Нет связи с пунктом (א).

$G = (V, E)$ – **неориентированный** граф [גרף לא מכוון], представленный списком смежностей [סמיכויות]:

a	→ b → c →
b	→ a → c → d → e →
c	→ d → a → b →
d	→ c → b → e →
e	→ b → d →

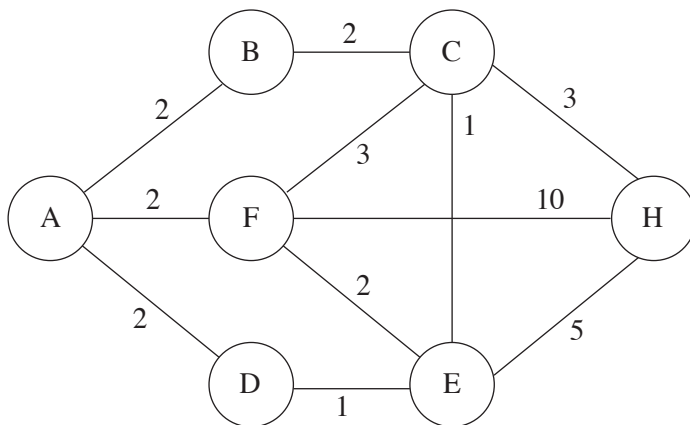
- (1) Начертите граф G , представленный данным списком.
- (2) Сколько компонентов связности (Connected Components) [רכיבי קשירות] есть в данном графе и каковы они?
- (3) Примените алгоритм сканирования в глубину (DFS) для данного графа G , начиная с вершины a .
Начертите в вашей тетради только полученное покрывающее дерево [העץ הפורש] (DFS) / покрывающий лес [היער הפורש] (DFS).
Основывайтесь на представлении, данном в списке смежностей.
- (4) Примените алгоритм сканирования в ширину (BFS) для данного графа G , начиная с вершины a .
Начертите в вашей тетради только полученное покрывающее дерево [העץ הפורש] (BFS) / покрывающий лес [היער הפורש] (BFS).
Основывайтесь на представлении, данном в списке смежностей.

/продолжение на странице 22/

11. В данном вопросе два не связанных между собой пункта, (א) и (ב).

Ответьте на вопросы обоих пунктов.

(א) Перед вами сеть $G = (V, E)$, где V – множество вершин сети, а E – множество ребер сети. Функция веса $w: E \rightarrow \mathbb{R}^+$ определяет вес (число, большее, чем 0) для каждого ребра в сети.



- (1) Найдите все самые короткие маршруты от вершины А до вершины Н в данной сети. Схематически изобразите каждый из этих маршрутов.
- (2) Найдите минимальное покрывающее дерево [העץ המכסה] в соответствии с алгоритмом Крускала, среди всех покрывающих деревьев, содержащих два данных ребра: (А, В) и (В, С). Схематически изобразите в своей тетради это минимальное покрывающее дерево.

(ב) Нет связи с пунктом (א).

Дана сеть $G = (V, E)$, где V – множество вершин сети,
а E – множество ребер сети. Функция веса $w: E \rightarrow \mathbb{R}^+$ определяет вес
(число, большее, чем 0) для каждого ребра в сети.

Кроме того, дано ребро $e = (a, b)$ и вершины s и t в данной сети.
Перед вами алгоритм, который возвращает значение 1, если данное
ребро e входит во все самые короткие маршруты от вершины s
до вершины t . Иначе – алгоритм возвращает значение 0.

В алгоритме отсутствуют два выражения, обозначенные (1) и (2).

Алгоритм:

Шаг 1: Применить алгоритм Дейкстры к сети G , и сохранить
в переменной $M1$ длину самого короткого маршрута
от вершины s до вершины t .

Шаг 2: Построить новую сеть $G1 = (V1, E1)$, где $V1 = V$, $E1 = \underline{\hspace{1cm}}(1)\underline{\hspace{1cm}}$

Шаг 3: Применить алгоритм Дейкстры к сети $G1$
и сохранить в переменной $M2$ длину самого короткого
маршрута от вершины s до вершины t .

Шаг 4: Если $\underline{\hspace{1cm}}(2)\underline{\hspace{1cm}}$, вернуть 1.

Иначе – вернуть 0.

Запишите в вашей тетради числа (1) и (2), и возле каждого из них
запишите недостающее выражение, которое оно обозначает.

12. В данном вопросе пять не связанных между собой пунктов (א)-(ה).
Ответьте на вопросы всех этих пунктов.

(א) В приведенной таблице дана часть возможного базового решения транспортной задачи [בעיית תובלה]: $x_{11} = 10$.

Пункты отправления	Пункты назначения				Предложение
	1	2	3	4	
1	20 10	20	30	21	20
2	22	17	29	30	30
3	10	24	26	38	10
Спрос	10	20	20	10	

- (1) Перепишите эту таблицу в тетрадь и дополните в ней возможное базовое решение в соответствии с методом северо-западного угла.
 (2) Цену в ячейке (3, 1) меняют с 10 на 20. Изменится ли возможное базовое решение найденное вами для подпункта (1)? Обоснуйте.
 (ג) В приведенной таблице дано возможное базовое решение транспортной задачи и дано значение u_3 .

Пункты отправления	Пункты назначения			Предложение	u_i
	1	2	3		
1	10	8	3 10	10	
2	12	15	9 3	3	
3	2 20	7 40	1 30	90	0
Спрос	20	40	43		
v_j					

(1) Перепишите эту таблицу в тетрадь и дополните в ней значения u_1, u_2, v_1, v_2, v_3 .

(2) Объясните, почему данное решение не является оптимальным.

Обратите внимание: продолжение вопроса на следующей странице.

/продолжение на странице 25/

- (ג) В приведенной ниже таблице дана транспортная задача
и даны значения $u_1, u_2, u_3, v_1, v_2, v_3$.

Пункты отправле- ния	Пункты назначения			Предло- жение	u_i
	1	2	3		
1	8	8	1	30	1
2	10	12	6	40	6
3	2	7	2	74	1
Спрос	30	64	50		
v_j	1	6	0		

- (1) Перепишите эту таблицу в тетрадь и запишите в ней возможное базовое решение, учитывая значения $u_1, u_2, u_3, v_1, v_2, v_3$.
- (2) В данной таблице меняют цену доставки из пункта отправления 1 в пункт назначения 2 с 8 на 7.
Учитывая это изменение, определите, какое из приведенных ниже утверждений i-iv верно, и перепишите его в вашу тетрадь. Обоснуйте свой выбор.

- i Базовое решение не изменится, и останется единственным решением, соответствующим данным значениям u_i и v_j .
- ii Есть **только** два возможных базовых решения, соответствующих данным значениям u_i и v_j .
- iii Есть бесконечное множество возможных базовых решений, соответствующих данным значениям u_i и v_j .
- iv Учитывая внесенное изменение, невозможно найти возможное базовое решение, соответствующее данным значениям u_i и v_j .

Обратите внимание: продолжение вопроса на следующей странице.
/продолжение на странице 26/

- (7) В приведенной ниже таблице дано возможное базовое решение транспортной задачи и даны значения $u_1, u_2, u_3, v_1, v_2, v_3$, соответствующие этому решению.

Пункты отправле- ния	Пункты назначения			Предло- жение	u_i
	1	2	3		
1	10	8 10	3	10	1
2	12	15	9 3	3	8
3	2 20	7 30	1 40	90	0
Спрос	20	40	43		
v_j	2	7	1		

Определите, какое из приведенных ниже утверждений i-iv верно, и перепишите его в вашу тетрадь. Обоснуйте свой выбор.

- i Данное решение не является возможным базовым решением.
- ii Данное решение является возможным базовым, но не является оптимальным решением.
- iii Данное решение является единственным оптимальным решением.
- iv Данное решение является оптимальным решением, но не является единственным оптимальным решением.

Обратите внимание: продолжение вопроса на следующей странице.

- (ה) В приведенной ниже таблице дано неоптимальное решение, полученное после k итераций для некоторой транспортной задачи, а также даны значения $u_1, u_2, u_3, v_1, v_2, v_3, v_4$.

Пункты отправле- ния	Пункты назначения				Предло- жение	u_i
	1	2	3	4		
1	4 6	4 4	14	25	10	3
2	6	1 5	13 10	14 3	18	0
3	0	8	14	12 2	2	-2
Спрос	6	9	10	5		
v_j	1	1	13	14		

Вы должны произвести дополнительную итерацию, то есть итерацию $k + 1$.

- (1) Какая переменная выходит из базы в этой итерации?
- (2) Начертите в вашей тетради новую таблицу и запишите в ней решение, полученное после этой итерации.

Вычислительные модели [מודלים חישוביים]

Если вы изучали этот курс, ответьте на два из вопросов 13-16 (за каждый вопрос – 25 баллов).

13. Перед вами регулярные языки L_1 и L_2 :

$$L_1 = \{a^{2n} \mid n \geq 0\} \text{ над алфавитом } \{a\}$$

$$L_2 = \{b^{2n+1} \mid n \geq 0\} \text{ над алфавитом } \{b\}$$

Дан язык L над алфавитом $\{a, b\}$.

$$L = \{a^n b^k \mid n \geq 0, k \geq 0, n \text{ четное и } k \text{ нечетное или } n \text{ нечетное и } k \text{ четное}\}$$

(א) Докажите, что язык L является регулярным. В доказательстве пользуйтесь только языками L_1 и L_2 и свойствами замкнутости [תכונות סגירות].

(ב) Постройте недетерминированный конечный автомат, принимающий язык L .

14. Перед вами языки L_1, L_2, L_3, L_4 над алфавитом $\{0, 1\}$.

$$L_1 = \{w \mid w \text{ содержит последовательность } 010\}$$

$$L_2 = \{w \mid w \text{ не содержит последовательность } 00\}$$

$$L_3 = \{w \mid \text{число нулей (0) в } w \text{ не равно числу единиц (1) в } w\}$$

$$L_4 = \{w \mid \text{число нулей (0) в } w \text{ нечетное}\}$$

Перед вами утверждения (1)-(6).

Для каждого из утверждений определите, верно оно или неверно.

Если утверждение верно, объясните почему.

Если утверждение неверно, объясните почему или запишите слово, опровергающее данное утверждение, и объясните, почему это слово его опровергает.

(1) ε принадлежит языку $L_1 \cup L_3 \cup L_4$

(2) 00100 принадлежит языку $L_1 \cap \overline{L_4}$

(3) $\overline{L_2} = \{w \mid w \text{ содержит последовательность } 11\}$

(4) $L_1 \subseteq L_3$

(5) $L_4 = R(L_4)$

(6) $L_3 \cap L_4 = L_2$

/продолжение на странице 29/

15. Перед вами язык L над алфавитом $\{a, b, c\}$

$$L = \{a^{2n}c^kb^{3n+1} \mid n > 0, k > 0\}$$

(א) Запишите самое короткое слово языка L .

(ב) Постройте стек-автомат (автомат с магазинной памятью)

[אוטומט מחסנית], принимающий [מקבל] язык L .

16. Дан метод, написанный на языках Java и C#.

Метод получает три целых числа, больших, чем 0.

Java	C#
<pre>public static int foo(int x , int y , int z) { if ((x % 3) == 0) return x; if ((x % 3) == 1) return y; return z; }</pre>	<pre>public static int Foo(int x , int y , int z) { if ((x % 3) == 0) return x; if ((x % 3) == 1) return y; return z; }</pre>

Запишите машину Тьюринга, которая реализует данный метод.

Ввод [קלט] машины – это три числа x, y, z , которые получает данный метод. Ввод записан в начале ленты. Каждое число записано в унарной форме. Числа отделены друг от друга символом $\#$.

Например, если метод получает 2 для x , 3 для y и 1 для z , то лента будет выглядеть так:

⊢	1	1	#	1	1	1	#	1	△	△	△	
---	---	---	---	---	---	---	---	---	---	---	---	-------

пишущая/читающая головка

Вывод [פלט] – это значение, возвращаемое методом. Вывод будет записан на ленту в некотором месте в виде унарного значения между двумя символами $\$$.

/продолжение на странице 30/

Объектно-ориентированное программирование [תכנות מונחה עצמים]

Если вы изучали этот курс и пишете на языке Java, ответьте на два вопроса из вопросов 17-20 (за каждый вопрос – 25 баллов).

17. В данном вопросе два не связанных между собой пункта (א)-(ב).

Ответьте на вопросы обоих пунктов.

(א) Даны пять классов `B`, `X`, `Y`, `Z`, `Run`. Приведенные ниже подпункты (1)-(4) относятся к этим классам. Подпункты не связаны друг с другом. Ответьте на вопросы всех подпунктов.

(1) В каждом из классов `B`, `X`, `Y`, `Z`

определен метод `public void foo() {}`

и в классе `Run` определен метод `public void bar(Object g) {g.foo();}`

Будет ли получена ошибка компиляции [הידור]?

если да, объясните почему.

(2) Считайте, что классы `X`, `Y`, `Z` наследуют класс `B`.

В каждом из классов `X`, `Y`, `Z` определен метод `public void foo() {}`

и в классе `Run` определен метод `public void bar(B g) {g.foo();}`

Будет ли получена ошибка компиляции? Если да, объясните почему.

(3) Считайте, что классы `X`, `Y`, `Z` наследуют класс `B`.

В каждом из классов `B`, `X`, `Y`, `Z`

определен метод `public void foo() {}`

и в классе `Run` определен метод `public void bar(B g) {g.foo();}`

Будет ли получена ошибка компиляции? Если да, объясните почему.

(4) Считайте, что классы `X`, `Y`, `Z` наследуют класс `B`.

В каждом из классов `B`, `X`, `Y`, `Z`

определен метод `public void foo() {}`

и в классе `Run` определен метод `public void bar(Object g) {g.foo();}`

Будет ли получена ошибка компиляции? Если да, объясните почему.

/продолжение на странице 31/

(ב) Нет связи с пунктом (א).

Перед вами класс **Singer**, который наследует класс **Artist**, а также интерфейс **IPerform**, реализуемый классом **Singer**.

```
interface IPerform
{
    void act();
    int train();
}

public class Singer extends Artist implements IPerform
{
    private int hits;
    public Singer(String name, double sal)
    {
        super(sal, name);
        this.hits = 5;
    }
    public Singer(double sal, int hits)
    {
        super(sal, "Singer Name");
        this.hits = hits;
    }
    public Singer(int hits)
    {
        super(6532.6, "Some", "One");
        this.hits = hits;
    }
    public double value() {return this.hits * this.price();}
    public int getNum()   {return Artist.num; }
    public double calc(double d)   {return d * super.calc(10.2); }
    public void print()           {super.print(); System.out.println("Singer");}
    public void act()             {System.out.println("I am singing"); }
}
```

Напишите на языке Java заголовок класса **Artist**, а также его атрибуты [תכונות] и заголовки методов, требуемых данным классом **Singer** и интерфейсом **IPerform**. Нет необходимости реализовывать методы класса **Artist**. Нельзя изменять класс **Singer** и интерфейс **IPerform**.

/продолжение на странице 32/

18. В зоопарке содержатся следующие животные: львы Lion, утконосы Platypus, змеи Snake и черепахи Turtle.

Львы и утконосы являются млекопитающими, а змеи и черепахи - пресмыкающимися.

Самки всех пресмыкающихся, содержащихся в зоопарке, откладывают яйца. Самки утконосов тоже откладывают яйца.

Требуется построить компьютеризированную систему, описывающую всех животных в зоопарке. Система будет показывать иллюстрации и видео.

С этой целью были определены классы **Animal** – животное, **Mammal** – млекопитающее и **Reptile** – пресмыкающееся.

У каждого животного есть атрибут name – имя и методы void move() и void turn() .

У животных, которые откладывают яйца, есть атрибут numOfEggs – среднее количество откладываемых яиц, и метод int getNumOfEggs() .

(а) Начертите иерархию классов для зоопарка и, если нужно, определите интерфейсы в максимальном соответствии с принципами объектно-ориентированного программирования (инкапсуляция [הכנסה] – encapsulation, наследование [הורשה] – inheritance, полиморфизм – polymorphism).

Укажите на чертеже связи между классами, а также связи между классами и интерфейсами, если таковые имеются.

Не нужно указывать на чертеже атрибуты и методы.

(б) Для каждого из классов на чертеже напишите на языке Java заголовок класса и его атрибуты, а также заголовки всех методов. Для каждого из интерфейсов на чертеже напишите на языке Java его заголовок и заголовки всех его методов.

Нет необходимости записывать методы get и set , а также конструкторы классов.

- (ג) В каждом из классов был определен метод `String details()` . Этот метод возвращает строку, состоящую из имени животного, перед которым указаны названия всех классов в иерархии наследования данного животного, начиная с класса **Animal**.

Например:

Для объекта `ln` класса **Lion**, с именем `Simba`, метод `details` вернет строку:

`Animal , Mammal , Lion , Simba`

Для объекта `plr` класса **Platypus**, с именем `Perry`, метод `details` вернет строку:

`Animal , Mammal , Platypus , Perry`

Реализуйте [ממש] на языке Java метод `details` в классах **Animal**, **Mammal**, **Lion**, **Platypus** так, чтобы метод возвращал требуемые строки. Реализуйте метод в максимальном соответствии с принципами объектно-ориентированного программирования (инкапсуляция [הכמסה] – *encapsulation*, наследование [הורשה] – *inheritance*, полиморфизм – *polymorphism*).

- (ד) Метод `toString` вызывает метод `details` . Если животное откладывает яйца, метод `toString` добавляет к строке слова `Number of eggs` и среднее количество откладываемых яиц и возвращает полученную строку.

Реализуйте [ממש] на языке Java метод `toString` в каждом из классов, где он определен, в максимальном соответствии с принципами объектно-ориентированного программирования (инкапсуляция [הכמסה] – *encapsulation*, наследование [הורשה] – *inheritance*, полиморфизм – *polymorphism*).

19. Перед вами классы **MyBills** , **Shopping** , **PhoneBill** , **Bill** и главный метод класса **Tester** . Проследите за выполнением главного метода и запишите полученный вывод [פלט].

Трассировка [מעקב] должна содержать значения переменных, а также для каждого объекта класса [פזו] значения его атрибутов.

```
public class Bill {
    public final static char DOLLAR = 'd';
    public final static char SHEKEL = 's';
    private static double dollarRate = 4.00;
    private double total;
    private char curr;
    public double getTotal() {
        if (curr == DOLLAR)
            return total * dollarRate;
        return total;
    }
    public Bill(double total, char curr) {
        this.total = total;
        this.curr = curr;
    }
    public String toString() {
        String s = getTotal() + " Shekels";
        return s;
    }
}

public class PhoneBill extends Bill {
    private String company;
    public PhoneBill(String company, double value, char curr) {
        super(value, curr);
        this.company = company;
    }
    public PhoneBill(PhoneBill pb) {
        super(pb.getTotal(), SHEKEL);
        company = pb.company;
    }
    public String toString() {
        String s = "Pay: " + super.toString() + " to: " + company;
        return s;
    }
}
```

/продолжение на странице 35/

```
public class Shopping extends Bill {
    private String name;
    private String list;
    public Shopping(String name, String list, double value, char curr) {
        super(value, curr);
        this.name = name;
        this.list = list;
    }
    public Shopping(Shopping s) {
        super(s.getTotal(), SHEKEL);
        name = s.name;
        list = s.list;
    }
    public String toString() {
        String s = name + " bought: " + list + ". Pay: " + super.toString();
        return s;
    }
}
```

```
public class MyBills {
    private int num;
    private Bill[] bills;
    private final static int MAX = 4;
    public MyBills() {
        bills = new Bill[MAX];
        num = 0;
    }
    public void addBill(Bill b) {
        if (num < MAX) {
            if (b instanceof PhoneBill)
                bills[num] = new PhoneBill((PhoneBill) b);
            else if (b instanceof Shopping)
                bills[num] = new Shopping((Shopping) b);
            else num--;
            num++;
        }
    }
}
```

Обратите внимание: продолжение вопроса на следующей странице.

/продолжение на странице 36/

```
public double total() {
    double t = 0;
    for (int i = 0; i < num; i++)
        t = t + bills[i].getTotal();
    return t;
}

public void print() {
    for (int i = 0; i < num; i++)
        System.out.println(bills[i].toString());
}

}

public class Tester {
    public static void main(String[] args) {
        MyBills baba = new MyBills();
        Shopping s = new Shopping("Baba", "Books", 250, Bill.DOLLAR);
        PhoneBill t = new PhoneBill("TeleBaba", 2000, Bill.SHEKEL);
        PhoneBill u = new PhoneBill("Galil Telecom", 200, Bill.DOLLAR);
        baba.addBill(t);
        baba.addBill(s);
        baba.addBill(u);
        baba.print();
        System.out.println(baba.total());
    }
}
```

20. Перед вами классы A , B .

Три пункта (а)-(г) относятся к этим классам, но не связаны друг с другом.
Ответьте на вопросы всех трех пунктов.

```
public class A {
    private int x;
    public A() { this.x = 0; }
    public A(int x) { this.x = x; }
    public int getX() { return x; }
    public void doubleX() { this.x = 2 * getX(); }
    public void tripleX() { this.x = 3 * getX(); }
    public void sub() { this.x = x - 1; }
    public void calc() { sub(); }
    public String toString() { return "xA="+this.x; }
}

public class B extends A {
    private int x;
    public B() { super(); this.x = 1; }
    public B(int x) { super(x); this.x = -x; }
    public B(int xA, int xB) { super(xA); this.x = xB; }
    public int getX() { return x; }
    public int baseX() { return super.getX(); }
    public void tenTimesX() { this.x = 10 * getX(); }
    public void tripleX() { this.x = 3 * getX(); }
    public void sub() { this.x = x - 2; }
    public String toString() { return super.toString()+" xB="+this.x+";" }
}
```

Обратите внимание: пункты этого вопроса на следующей странице.

- (א) Перед вами серия команд, вывод [פלט] которой должен выглядеть так: xA=1 xB=20; xA=1 xB=20; xA=1 xB=20; xA=1 xB=20;

В данной серии команд есть ошибка компиляции [הידור]. Исправьте эту ошибку, чтобы получить требуемый вывод.

```
A a1 = new B(1, 20);  
Object obj = a1;  
B b1 = a1;  
A a2 = a1;  
System.out.println(a1+" "+obj+" "+a2+" "+b1);
```

- (ב) Перед вами серия команд. Покажите объекты классов, создаваемые в процессе их выполнения, и для каждого объекта класса отобразите значения его атрибутов.

```
A aa = new B(3, 10);  
aa.sub();  
Object[] ar = new Object[6];  
ar[0] = new A();  
ar[1] = new A(5);  
ar[2] = new B();  
ar[3] = new B(5);  
ar[4] = new B(2, 4);  
ar[5] = aa;  
((A)ar[3]).tripleX();  
((B)ar[4]).tenTimesX();
```

(ג) Перед вами главный метод.

```
public static void main(String[] args)
{
    A a1 = new A(1);
    A a2 = new B(2, 99);
    /***
}
```

Перед вами фрагменты (i)-(vi).

- (i) a2.doubleX();
 System.out.println(a2);
- (ii) a2.tenTimesX();
 System.out.println(a2.tenTimesX());
- (iii) if (a2 instanceof B)
 {
 a2.tenTimesX();
 System.out.println(a2);
 }
- (iv) ((B)a1).tenTimesX();
 System.out.println(a1);
- (v) a2.calc();
 System.out.println(a2);
- (vi) B bb = (B)a2;
 System.out.println(bb.baseX());

Для каждого из фрагментов (i)-(vi) выполните следующее:

- Подставьте фрагмент вместо /*** в главном методе.
- Определите, является ли фрагмент корректным [תקין].
- Если фрагмент корректен, запишите вывод [פלט], получаемый в результате его выполнения.
Если фрагмент некорректен, запишите вид ошибки:
ошибка компиляции [הידור] или выполнения [ריצה].

Объектно-ориентированное программирование
[תכנות מונחה עצמים]

Если вы изучали этот курс и пишете на языке C#, ответьте на два вопроса из вопросов 21-24 (за каждый вопрос – 25 баллов).

21. В данном вопросе два не связанных между собой пункта (א)-(ב).

Ответьте на вопросы обоих пунктов.

(א) Даны пять классов B, X, Y, Z, Run. Приведенные ниже подпункты (1)-(4) относятся к этим классам. Подпункты не связаны друг с другом. Ответьте на вопросы всех подпунктов.

(1) В каждом из классов B, X, Y, Z

определен метод `public virtual void Foo(){}`

и в классе Run определен метод `public void Bar (Object g) {g.Foo(); }`

Будет ли получена ошибка компиляции [הידור]?

Если да, объясните почему.

(2) Считайте, что классы X, Y, Z наследуют класс B.

В каждом из классов X, Y, Z

определен метод `public virtual void Foo() { }`

и в классе Run определен метод `public void Bar(B g) {g.Foo(); }`

Будет ли получена ошибка компиляции? Если да, объясните почему.

(3) Считайте, что классы X, Y, Z наследуют класс B.

В каждом из классов B, X, Y, Z

определен метод `public virtual void Foo(){}`

и в классе Run определен метод `public void Bar(B g) {g.Foo(); }`

Будет ли получена ошибка компиляции? Если да, объясните почему.

(4) Считайте, что классы X, Y, Z наследуют класс B.

В каждом из классов B, X, Y, Z

определен метод `public virtual void Foo() { }`

и в классе Run определен метод `public void Bar(Object g) {g.Foo(); }`

Будет ли получена ошибка компиляции? Если да, объясните
/продолжение на странице 41/

(ב) Нет связи с пунктом (א).

Перед вами класс **Singer**, который наследует класс **Artist**, а также интерфейс **IPerform**, реализуемый классом **Singer**.

```
interface IPerform
{
    void Act();
    int Train();
}

public class Singer : Artist, IPerform
{
    private int hits;
    public Singer(string name, double sal) : base(sal, name)
    {
        this.hits = 5;
    }
    public Singer(double sal, int hits) : base(sal, "Singer Name")
    {
        this.hits = hits;
    }
    public Singer(int hits) : base(6532.6, "Some", "One")
    {
        this.hits = hits;
    }
    public double Value() {return this.hits * this.Price(); }
    public int GetNum() {return Artist.num; }
    public override double Calc(double d) {return d * base.Calc(10.2); }
    public override void Print() {base.Print(); Console.WriteLine("Singer"); }
    public void Act() {Console.WriteLine("I am singing"); }
}
}
```

Напишите на языке C# заголовок класса **Artist**, а также его атрибуты [תכונות] и заголовки методов, требуемых данным классом **Singer** и интерфейсом **IPerform**. Нет необходимости реализовывать методы класса **Artist**.

Нельзя изменять класс **Singer** и интерфейс **IPerform**.

22. В зоопарке содержатся следующие животные: львы Lion, утконосы Platypus, змеи Snake и черепахи Turtle.

Львы и утконосы являются млекопитающими, а змеи и черепахи - пресмыкающимися.

Самки всех пресмыкающихся, содержащихся в зоопарке, откладывают яйца. Самки утконосов тоже откладывают яйца.

Требуется построить компьютеризированную систему, описывающую всех животных в зоопарке. Система будет показывать иллюстрации и видео.

С этой целью были определены классы **Animal** – животное, **Mammal** – млекопитающее и **Reptile** – пресмыкающееся.

У каждого животного есть атрибут name – имя, и методы void Move() и void Turn() .

У животных, которые откладывают яйца, есть атрибут numOfEggs – среднее количество откладываемых яиц, и метод int GetNumOfEggs() .

(а) Начертите иерархию классов для зоопарка и, если нужно, определите интерфейсы в максимальном соответствии с принципами объектно-ориентированного программирования (инкапсуляция [הכמסה] – encapsulation, наследование [הורשה] – inheritance, полиморфизм – polymorphism).

Укажите на чертеже связи между классами, а также связи между классами и интерфейсами, если таковые имеются.

Не нужно указывать на чертеже атрибуты и методы.

(б) Для каждого из классов на чертеже напишите на языке C# заголовок класса и его атрибуты, а также заголовки всех методов.

Для каждого из интерфейсов на чертеже напишите на языке C# его заголовок и заголовки всех его методов.

Нет необходимости записывать методы Get и Set , а также конструкторы классов.

- (ג) В каждом из классов был определен метод `string Details()` . Этот метод возвращает строку, состоящую из имени животного, перед которым указаны названия всех классов в иерархии наследования данного животного, начиная с класса **Animal**.

Например:

Для объекта `ln` класса **Lion**, с именем `Simba`, метод `Details` вернет строку:

`Animal , Mammal , Lion , Simba`

Для объекта `plr` класса **Platypus**, с именем `Perry`, метод `Details` вернет строку:

`Animal , Mammal , Platypus , Perry`

Реализуйте [ממש] на языке C# метод `Details` в классах **Animal**, **Mammal**, **Lion**, **Platypus** так, чтобы метод возвращал требуемые строки. Реализуйте метод в максимальном соответствии с принципами объектно-ориентированного программирования (инкапсуляция [הכנסה] – encapsulation, наследование [הורשה] – inheritance, полиморфизм – polymorphism).

- (ד) Метод `ToString` вызывает метод `Details` . Если животное откладывает яйца, метод `ToString` добавляет к строке слова `Number of eggs` и среднее количество откладываемых яиц и возвращает полученную строку.

Реализуйте [ממש] на языке C# метод `ToString` в каждом из классов, где он определен, в максимальном соответствии с принципами объектно-ориентированного программирования (инкапсуляция [הכנסה] – encapsulation, наследование [הורשה] – inheritance, полиморфизм – polymorphism).

23. Перед вами классы **MyBills** , **Shopping** , **PhoneBill** , **Bill** и главный метод класса **Tester** . Проследите за выполнением главного метода и запишите полученный вывод [פלט].
Трассировка [מעקב] должна содержать значения переменных, а также для каждого объекта класса [פזו] значения его атрибутов.

```
public class Bill {
    public const char DOLLAR = 'd';
    public const char SHEKEL = 's';
    private static double dollarRate = 4.00;
    private double total;
    private char curr;
    public double GetTotal() {
        if (curr == DOLLAR)
            return total * dollarRate;
        return total;
    }
    public Bill(double total, char curr) {
        this.total = total;
        this.curr = curr;
    }
    public override string ToString() {
        string s = GetTotal() + " Shekels";
        return s;
    }
}
```

```
public class PhoneBill : Bill {
    private string company;
    public PhoneBill(string company, double value, char curr) : base(value, curr) {
        this.company = company;
    }
    public PhoneBill(PhoneBill pb) : base(pb.GetTotal(), SHEKEL){
        company = pb.company;
    }
    public override string ToString() {
        string s = "Pay: " + base.ToString() + " to: " + company;
        return s;
    }
}
```

/продолжение на странице 45/

```
public class Shopping : Bill {
    private string name;
    private string list;
    public Shopping(string name, string list, double value, char curr) : base(value, curr){
        this.name = name;
        this.list = list;
    }
    public Shopping(Shopping s):base (s.GetTotal(), SHEKEL) {
        name = s.name;
        list = s.list;
    }
    public override string ToString() {
        string s = name + " bought: " + list + ". Pay: " + base.ToString();
        return s;
    }
}

public class MyBills {
    private int num;
    private Bill[] bills;
    private const int MAX = 4;
    public MyBills() {
        bills = new Bill[MAX];
        num = 0;
    }
    public void AddBill(Bill b) {
        if (num < MAX) {
            if (b is PhoneBill)
                bills[num] = new PhoneBill((PhoneBill) b);
            else if (b is Shopping)
                bills[num] = new Shopping((Shopping) b);
            else num--;
            num++;
        }
    }
}
```

Обратите внимание: продолжение вопроса на следующей странице.

/продолжение на странице 46/

```
public double Total() {  
    double t = 0;  
    for (int i = 0; i < num; i++)  
        t = t + bills[i].GetTotal();  
    return t;  
}  
public void Print() {  
    for (int i = 0; i < num; i++)  
        Console.WriteLine(bills[i].ToString());  
}  
}
```

```
public class Tester {  
    public static void Main(string[] args) {  
        MyBills baba = new MyBills();  
        Shopping s = new Shopping("Baba", "Books", 250, Bill.DOLLAR);  
        PhoneBill t = new PhoneBill("TeleBaba", 2000, Bill.SHEKEL);  
        PhoneBill u = new PhoneBill("Galil Telecom", 200, Bill.DOLLAR);  
        baba.AddBill(t);  
        baba.AddBill(s);  
        baba.AddBill(u);  
        baba.Print();  
        Console.WriteLine(baba.Total());  
    }  
}
```

24. Перед вами классы А , В .

Три пункта (ж)-(з) относятся к этим классам, но не связаны друг с другом.

Ответьте на вопросы всех трех пунктов.

```
public class A {
    private int x;
    public A() { this.x = 0; }
    public A(int x) { this.x = x; }
    public virtual int GetX() { return x; }
    public void DoubleX() { this.x = 2 * GetX(); }
    public virtual void TripleX() { this.x = 3 * GetX(); }
    public virtual void Sub() { this.x = x - 1; }
    public void Calc() { Sub(); }
    public override string ToString(){ return "xA="+this.x; }
}

public class B:A {
    private int x;
    public B() : base() { this.x = 1; }
    public B(int x) : base(x) { this.x = -x; }
    public B(int xA, int xB) : base(xA) { this.x = xB; }
    public override int GetX() { return x; }
    public int BaseX() { return base.GetX(); }
    public void TenTimesX() { this.x = 10 * GetX(); }
    public override void TripleX() { this.x = 3 * GetX(); }
    public override void Sub() { this.x = x - 2; }
    public override string ToString(){ return base.ToString()+" xB="+this.x+";";}
}
```

Обратите внимание: пункты вопроса на следующей странице.

- (н) Перед вами серия команд, вывод [פלט] которой должен выглядеть так: xA=1 xB=20; xA=1 xB=20; xA=1 xB=20; xA=1 xB=20;

В данной серии команд есть ошибка компиляции. Исправьте эту ошибку, чтобы получить требуемый вывод.

```
A a1 = new B(1, 20);  
Object obj = a1;  
B b1 = a1;  
A a2 = a1;  
Console.WriteLine(a1+" "+obj+" "+a2+" "+b1);
```

- (д) Перед вами серия команд. Покажите объекты классов, создаваемые в процессе их выполнения, и для каждого объекта класса отобразите значения его атрибутов.

```
A aa = new B(3, 10);  
aa.Sub();  
Object[] ar = new Object[6];  
ar[0] = new A();  
ar[1] = new A(5);  
ar[2] = new B();  
ar[3] = new B(5);  
ar[4] = new B(2, 4);  
ar[5] = aa;  
((A)ar[3]).TripleX();  
((B)ar[4]).TenTimesX();
```


(ג) Перед вами главный метод.

```
public static void Main()
{
    A a1 = new A(1);
    A a2 = new B(2, 99);
    /***
}
```

Перед вами фрагменты (i)-(vi).

- (i) a2.DoubleX();
Console.WriteLine(a2);
- (ii) a2.TenTimesX();
Console.WriteLine(a2.TenTimesX());
- (iii) if (a2 is B)
{
a2.TenTimesX();
Console.WriteLine(a2);
}
- (iv) ((B)a1).TenTimesX();
Console.WriteLine(a1);
- (v) a2.Calc();
Console.WriteLine(a2);
- (vi) B bb = (B)a2;
Console.WriteLine(bb.BaseX());

Для каждого из фрагментов (i)-(vi) выполните следующее:

- Подставьте фрагмент вместо /*** в главном методе.
- Определите, является ли фрагмент корректным [תקין].
- Если фрагмент корректен, запишите вывод [פלט], получаемый в результате его выполнения.
- Если фрагмент некорректен, запишите вид ошибки: ошибка компиляции [הידור] или выполнения [ריצה].

Желаем успеха!

Авторские права принадлежат Государству Израиль.
Копировать или публиковать можно только
с разрешения Министерства просвещения.

בהצלחה!

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