

Обратите внимание: в этом вопроснике есть специальные инструкции.
Отвечайте на вопросы, следуя этим инструкциям.

שימו לב: בבחינה זו יש הנחיות מיוחדות.
יש לענות על השאלות על פי הנחיות אלה.

Компьютеры

מדעי המחשב

Указания

הוראות

- а. Продолжительность экзамена: три часа.
- б. Строение вопросника и ключ к оценке:
в этом вопроснике три раздела.
- | | | | | |
|----------------|---------------------------------|---|-----|--------|
| Раздел первый: | $(1 \times 10) + (1 \times 15)$ | – | 25 | баллов |
| Раздел второй: | (2×25) | – | 50 | баллов |
| Раздел третий: | (1×25) | – | 25 | баллов |
| Всего | | – | 100 | баллов |

- א. משך הבחינה: שלוש שעות.
- ב. מבנה השאלון ומפתח ההערכה:
בשאלון זה שלושה פרקים.
- | | | | | |
|-----------|---------------------------------|---|-----|--------|
| פרק ראשון | $(10 \times 1) + (15 \times 1)$ | – | 25 | נקודות |
| פרק שני | (25×2) | – | 50 | נקודות |
| פרק שלישי | (25×1) | – | 25 | נקודות |
| סך הכול | | – | 100 | נקודות |

- в. Разрешенный вспомогательный материал:
любой вспомогательный материал, за исключением калькулятора с возможностью программирования.

- ג. חומר עזר מותר בשימוש:
כל חומר עזר, חוץ ממחשבון שיש בו אפשרות תכנות.

- г. Особые указания:

- ד. הוראות מיוחדות:

- (1) **На внешней обложке тетради укажите название курса, который вы изучали.**
Курс – это один из из четырех следующих курсов: “Компьютерные системы и ассемблер”, “Алгоритмы”, “Вычислительные модели”, “Объектно-ориентированное программирование”.

- (1) **רשמו על הכריכה החיצונית של המחברת את שם המסלול שלמדתם.**
המסלול הוא אחד מארבעת המסלולים האלה:
מערכות מחשב ואסמבלי, אלגוריתמים, מודלים חישוביים, תכנות מונחה עצמים.

- (2) **Все программы, которые вы должны написать на языке программирования в первом и втором разделах, пишите только на одном языке – Java или C# .**

- (2) את כל התוכניות שיש לכתוב בשפת מחשב בפרקים הראשון והשני כתבו בשפה אחת בלבד – Java או C# .

Примечание: баллы сниматься не будут, если в написанных вами программах вы напишете большую букву вместо маленькой или наоборот.

הערה: לא יורדו נקודות אם תכתבו בתוכניות אות גדולה במקום אות קטנה או להפך.

יש לכתוב במחברת הבחינה בלבד. יש לרשום “טייטה” בראש כל עמוד המשמש טיוטה.
כתיבת טיוטה בדפים שאינם במחברת הבחינה עלולה לגרום לפסילת הבחינה.

Пишите только в экзаменационной тетради. Напишите слово «טייטה» 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!

В о п р о с ы

В этом вопроснике три раздела.

Вы должны ответить на вопросы трех этих разделов, согласно инструкциям каждого раздела.

Примечание: в каждом вопросе, в котором требуется ввод,

нет необходимости проверять корректность вводимых данных.

Для решающих на языке Java: в каждом вопросе, в котором требуется ввод,

считайте, что в программе написана команда:

```
Scanner input = new Scanner (System.in);
```

РАЗДЕЛ ПЕРВЫЙ (25 баллов)

Ответьте на вопрос 1 – обязательный (10 баллов).

1. Напишите внешний метод [פעולה חיצונית] multiply на языке Java или Multiply на языке C#, получающий два массива целого типа, arr1 и arr2, различные по длине. Обозначим длину меньшего массива k и обозначим длину большего массива m. Метод должен вернуть новый массив целого типа длиной m (длина большего из двух массивов arr1 и arr2).

Полученный массив должен соответствовать следующему описанию:

Значение каждой из первых k ячеек полученного массива равно произведению значений в соответствующих ей ячейках массивов arr1 и arr2.

Остальные значения ячеек полученного массива равны значениям соответствующих им ячеек в большем массиве.

Пример: для массивов arr1 и arr2, приведенных ниже:

	0	1	2	3	4
arr1	1	- 4	4	9	2

	0	1	2	3	4	5	6
arr2	9	2	0	- 1	3	11	23

метод вернет массив, который выглядит так:

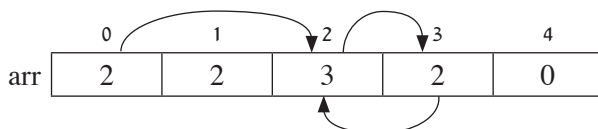
	0	1	2	3	4	5	6
	9	- 8	0	- 9	6	11	23

Примечание: считайте, что в двух этих массивах нет пустых ячеек и значения массивов корректны.

Ответьте на один из вопросов 2–3 (15 баллов).

2. "Сканирование в соответствии со значениями ячеек" – это сканирование массива, которое начинается в индексе 0, и переходит на индекс, равный значению текущей просканированной ячейки (ячейка с индексом 0), и оттуда тем же способом продолжает сканирование следующей ячейки, и так далее. Если одно из значений ячеек равно 0 и сканирование приходит в нее, оно заканчивается в ней, иначе сканирование не заканчивается.

Пример:



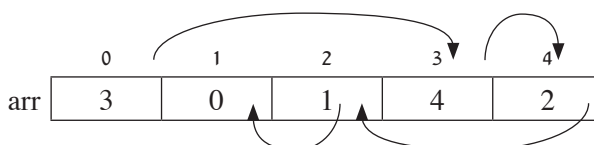
Порядок индексов в этом сканировании: 0, 2, 3, 2, 3, ...

Считайте, что значения ячеек не выходят за пределы массива.

Законченный массив – это массив, в котором при "сканировании в соответствии со значениями ячеек" выполняется и первое условие, и второе условие, приведенные ниже:

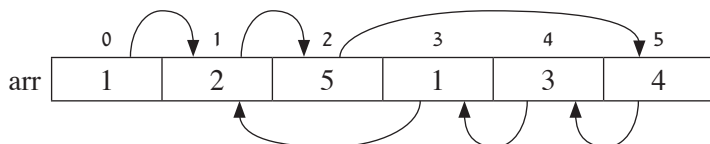
1. **Все** ячейки массива сканируются.
2. Сканирование заканчивается (потому что в массиве есть ячейка со значением 0 и сканирование доходит до нее).

Пример законченного массива:



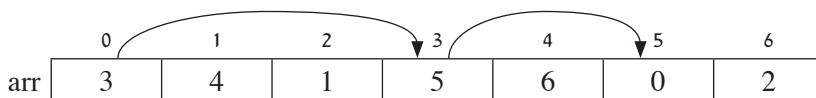
Массив в этом примере – это законченный массив, потому что сканирование проходит по всем ячейкам и заканчивается.

Пример массива, который **не** является законченным:



Этот массив не является законченным массивом, потому что сканирование не заканчивается.

Дополнительный пример массива, который **не** является законченным:



Этот массив не является законченным массивом, так как сканирование заканчивается, а ячейки с индексами 1, 2, 4, 6 остаются непросканированными.

Реализуйте метод:

Java – public static boolean isPerfect (int[] arr)

C# – public static bool IsPerfect (int[] arr)

Метод вернет true, если полученный массив `arr` является законченным, иначе метод вернет false.

Примечание: нет необходимости сохранять значения массива.

Считайте, что значения ячеек не выходят за границы массива. /продолжение на странице 4/

3. Пиксель (Pixel) – это единица информации, описывающая точку на дигитальном изображении.

Каждый пиксель состоит из сочетания трех цветов: красного, зеленого и синего.

Каждый из трех этих цветов (красный, зеленый и синий) получает численное значение от 0 до 255 (включительно). Число представляет силу цвета в точке. Посредством сочетания трех цветов в различных значениях от 0 до 255 можно получить любой оттенок цветовой палитры.

Для получения точки, цвет которой красный (между светло-красным и темно-красным) значения цветов должны быть:

красный – от 1 до 255, зеленый – 0, синий – 0.

Для получения точки, цвет которой зеленый (между светло-зеленым и темно-зеленым) значения цветов должны быть:

красный – 0, зеленый – от 1 до 255, синий – 0.

Для получения точки, цвет которой синий (между светло-синим и темно-синим) значения цветов должны быть:

красный – 0, зеленый – 0, синий – от 1 до 255.

Для получения точек остальных цветов (не включая черного и белого) значения по меньшей мере двух из цветов красный, зеленый и синий должны быть больше, чем 0.

Для получения точки, цвет которой черный, значения цветов красный, зеленый и синий должны быть 0.

Для получения точки, цвет которой белый, значения цветов красный, зеленый и синий должны быть 255.

Дан класс **Pixel** – Пиксель с тремя атрибутами [תכונות]:

red – красный цвет, целое число от 0 до 255

green – зеленый цвет, целое число от 0 до 255

blue – синий цвет, целое число от 0 до 255

Считайте, что определены методы get/Get и set/Set для этих атрибутов класса.

Перед вами частичный интерфейс класса Pixel :

Заголовок метода	Описание метода
public Pixel (int red, int green, int blue)	Конструктор [פעולה בונה], получающий значения для атрибутов класса
Java – public boolean isRed () C# – public bool IsRed ()	Метод возвращает true, если точка красного цвета, иначе метод возвращает false
Java – public boolean isGreen () C# – public bool IsGreen ()	Метод возвращает true, если точка зеленого цвета, иначе метод возвращает false
Java – public boolean isBlue () C# – public bool IsBlue ()	Метод возвращает true, если точка синего цвета, иначе метод возвращает false

Обратите внимание: продолжение вопроса на следующей странице.

/продолжение на странице 5/

- (8) Реализуйте метод `isRed/IsRed` .

Считайте, что значения атрибутов корректны (от 0 до 255).

- (9) В этом пункте можно пользоваться методами `isGreen / IsGreen` и `isBlue / IsBlue` , не реализуя их.

Можно добавить дополнительные методы в класс `Pixel` , но необходимо реализовать их.

Дан класс **Structure** , в котором только один атрибут:

`arr` – одномерный массив типа **Pixel** длиной `N` .

- (1) Напишите внутренний метод [פונקציה פנימית] класса `Structure` с именем `isBalanced` на языке Java или с заголовком `IsBalanced` на языке C# .

Метод вернет `true` , если в массиве `arr` число точек красного цвета равно числу точек зеленого цвета и равно числу точек синего цвета. Иначе метод вернет `false` .

Обратите внимание: метод вернет `true` , даже если есть точки других цветов, а не красного, зеленого и синего, пока число красных, зеленых и синих точек одинаково.

- (2) "Черно-белый" массив – массив, в котором все точки только черного и белого цвета и в котором есть по меньшей мере одна точка белого цвета и по меньшей мере одна точка черного цвета.

Напишите внутренний метод в классе `Structure` с заголовком `isBlackWhite` на языке Java или `IsBlackWhite` на языке C# . Метод вернет `true` , если массив `arr` – это "черно-белый" массив, иначе метод вернет `false` .

РАЗДЕЛ ВТОРОЙ (50 баллов)

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

Ответьте на два из вопросов 4–7 (за каждый вопрос – 25 баллов).

4. Дан класс **Range** – интервал, в котором два атрибута [תכונות]:

- low – число целого типа
- high – число целого типа

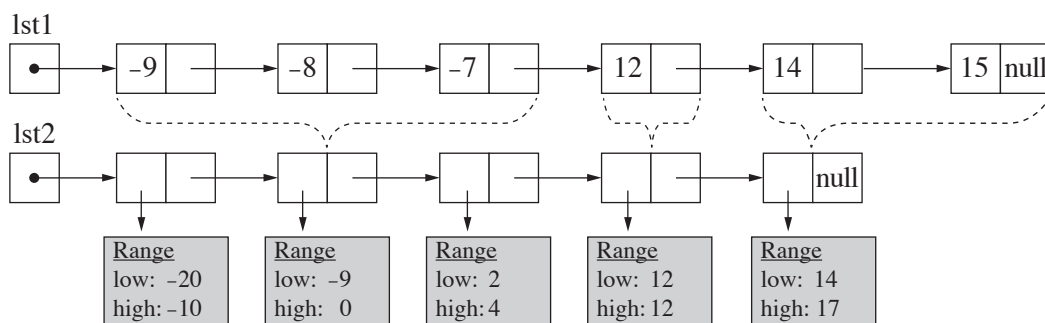
Число high больше или равно low ($high \geq low$).

Считайте, что определены методы get/Get и set/Set для атрибутов этого класса.

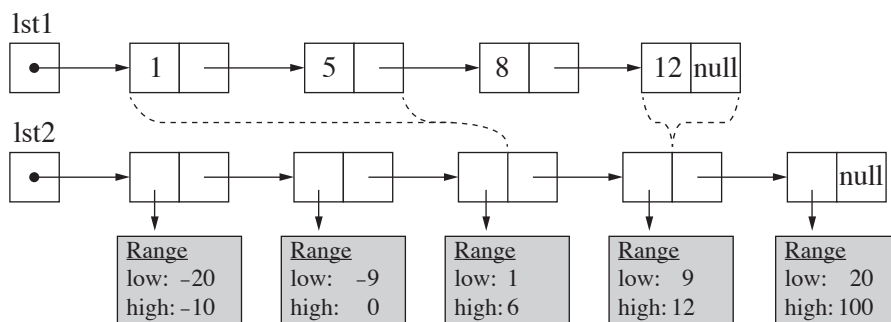
Некоторое число x включено в объект типа Range, если оно находится в интервале чисел между low и high ($high \geq x \geq low$).

Цепочка звеньев [שרשרת חוליות] – lst1 целого типа включена в цепочку звеньев lst2 типа Range, если для каждого числа в цепочке lst1 существует звено в цепочке lst2, которое его включает.

Пример цепочки lst1, которая включена в цепочку lst2:



Пример цепочки lst1, которая не включена в цепочку lst2:



Объяснение: число 8 в цепочке lst1 не включено ни в какое звено цепочки lst2.

Обратите внимание: продолжение вопроса на следующей странице.

Реализуйте внешний метод [פעולה חיצונית], приведенный ниже:

Java – public static boolean isIncluded (Node<Integer> lst1, Node<Range> lst2)

C# – public static bool IsIncluded (Node<int> lst1, Node<Range> lst2)

Метод вернет true , если lst1 включена в lst2 , иначе метод вернет false. Метод должен работать со временной сложностью выполнения [סיבוכיות זמן הריצה] $O(N)$.

Примечание: N – длина более длинной из двух этих цепочек.

Допущения:

- lst1 и lst2 не null .
- в цепочке lst2 все объекты типа Range не null.
- **цепочка lst1 отсортирована в порядке возрастания.**
- **цепочка lst2 отсортирована в порядке возрастания**, т.е. значение high каждого звена меньше значения low следующего после него в цепочке звена (как в примерах выше).

5. Дан класс **TwoStack** с двумя атрибутами:

- numbers – стек [מחסנית] целого типа
- sums – стек целого типа

Соотношение между стеком sums и стеком numbers описывается так:

элемент в хвосте стека sums равен элементу в хвосте стека numbers ,
второй от хвоста элемент стека sums равен сумме двух последних элементов стека numbers ,
третий от хвоста элемент стека sums равен сумме трех последних элементов стека numbers ,
и так далее до элемента в голове стека sums , равного сумме всех чисел в стеке numbers .

Пример:

	sums		numbers	
2+(-1)+4+3+(-9)	-1	← голова стека	-9	← голова стека
2+(-1)+4+3	8		3	
2+(-1)+4	5		4	
2+(-1)	1		-1	
2	2	← хвост стека	2	← хвост стека

(*) Реализуйте внутренний метод [פעולה פנימית], приведенный ниже:

Java – public Stack<Integer> getNums (int x)

C# – public Stack<int> GetNums (int x)

Метод получает число x , равное одному из чисел стека sums, и возвращает новый стек целого типа, содержащий числа из стека numbers , суммы которых равны числу x .

Считайте, что число x существует в стеке sums и появляется в нем только один раз.

Пример:

Для стеков в приведенном выше примере и для x = 5 (5 – третье число снизу в стеке sums), метод вернет следующий стек:

4	← голова стека
-1	
2	

Примечания: – Можно изменять стеки этого класса.

– Порядок чисел в возвращаемом стеке не имеет значения.

– Определение стека целого типа Stack<int>/Stack<Integer> равноценно определению StackInt . Можно пользоваться любым из них, по вашему выбору.

Обратите внимание: продолжение вопроса на следующей странице.

(ב) Реализуйте внутренний метод [פעולה פנימית], приведенный ниже:

Java – public void eraseNum (int x)

C# – public void EraseNum (int x)

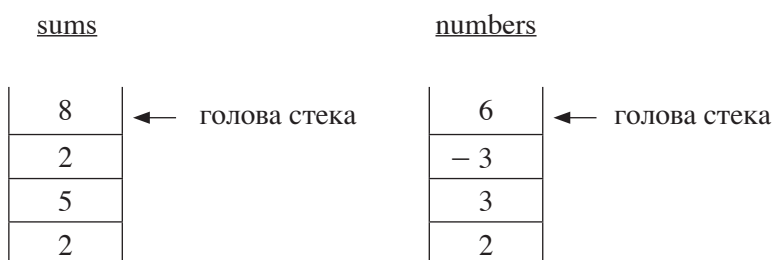
Метод удаляет число x из стека numbers и исправляет стек sums , соответственно.

Считайте, что число x существует в стеке numbers и появляется в нем только один раз.

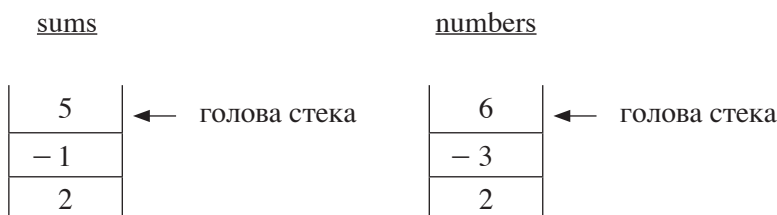
Примечание: необходимо сохранять порядок чисел, оставшихся в стеке numbers .

Пример:

Для стеков в приведенном ниже примере и для $x = 3$:



После выполнения этого метода стеки будут выглядеть так:



6. **Обратите внимание! Этот вопрос сформулирован дважды:**
на языке Java на страницах 10–11 и на языке C# на страницах 12–13.

На языке Java

(*) Дан метод `stackSod1` и стек `st` :

6	← голова стека
3	
7	
4	

```
public static void stackSod1(Stack<Integer> st, int element)
{
    if(st.isEmpty())
        st.push(element);
    else
    {
        int val = st.pop();
        stackSod1(st, element);
        st.push(val);
    }
}
```

Примечание: определение стека [מחסנית] целого типа `Stack<Integer>` равноценно определению `StackInt` .

- (1) Начертите стек, как он будет выглядеть после вызова метода `stackSod1(st, 9)` .
Необходимо показать трассировку [מעקב].
- (2) Какова цель метода `stackSod1` ?
- (3) Какова временная сложность выполнения [סיבוכיות זמן הריצה] метода `stackSod1` ?

Обратите внимание: продолжение вопроса на следующей странице.

(א) Перед вами метод `stackSod2` и стек `st` :

6	← голова стека
3	
7	
4	

```
public static void stackSod2 (Stack<Integer> st)
{
    if(!st.isEmpty())
    {
        int val = st.pop();
        stackSod2(st);
        stackSod1(st, val);
        st.push(val);
    }
}
```

(1) Начертите стек, как он будет выглядеть после вызова метода `stackSod2(st)` .

Необходимо показать трассировку [מעקב].

Примечание: в этом пункте не нужно выполнять трассировку метода `stackSod1` .

(2) Какова цель метода `stackSod2` ?

(3) Какова временная сложность выполнения [סיבוכיות זמן הריצה] метода `stackSod2` ?

На языке C#

(*) Дан метод StackSod1 и стек st :

6	← голова стека
3	
7	
4	

```
public static void StackSod1(Stack<int> st, int element)
{
    if(st.IsEmpty())
        st.Push(element);
    else
    {
        int val = st.Pop();
        StackSod1(st, element);
        st.Push(val);
    }
}
```

Примечание: определение стека целого типа Stack<int> равноценно
определению StackInt .

- (1) Начертите стек, как он будет выглядеть после вызова метода StackSod1(st, 9) .
Необходимо показать трассировку [מעקב].
- (2) Какова цель метода StackSod1 ?
- (3) Какова временная сложность выполнения [סיבוכיות זמן הריצה] метода StackSod1 ?

Обратите внимание: продолжение вопроса на следующей странице.

(א) Перед вами метод StackSod2 и стек st :

6	← голова стека
3	
7	
4	

```
public static void StackSod2 (Stack<int> st)
{
    if(!st.IsEmpty())
    {
        int val = st.Pop();
        StackSod2(st);
        StackSod1(st, val);
        st.Push(val);
    }
}
```

- (1) Начертите стек, как он будет выглядеть после вызова метода StackSod2(st) .
Необходимо показать трассировку [מעקב].
Примечание: в этом пункте не нужно выполнять трассировку метода StackSod1 .
- (2) Какова цель метода StackSod2 ?
- (3) Какова временная сложность выполнения [סיבוכיות זמן הריצה] метода StackSod2 ?

7. Вопрос на тему бинарного дерева

В этом вопросе вы можете пользоваться внешним методом [פעולה חיצונית] eraseFirst / EraseFirst , приведенным ниже, не реализуя его.

Заголовок метода	Описание метода	Примеры
На языке Java : <pre>public static String eraseFirst (String str)</pre> На языке C# : <pre>public static string EraseFirst (string str)</pre>	Метод возвращает подстроку строки str , без первого знака. Если строка – str пуста до вызова метода, будет ошибка.	– Для строки str = "hello" , метод вернет строку "ello". – Для строки str = "temp" , метод вернет строку "emp". – Для строки str = "m" , метод вернет пустую строку "". – Для пустой строки str = "" , будет ошибка.

Реализуйте приведенный ниже внешний метод:

Java – public static boolean wordFromRoot (BinNode<Character> tree, String str)

C# – public static bool WordFromRoot (BinNode<char> tree, string str)

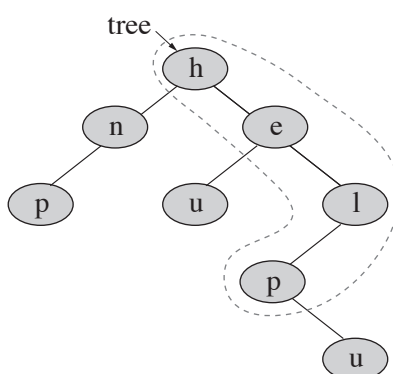
Метод принимает строку – str , которая содержит по меньшей мере один знак, и указатель на бинарное дерево знаков – tree , которое не null .

Метод вернет true , если существует маршрут, начинающийся в корне дерева, в котором последовательность знаков идентична строке – str . Иначе метод вернет false .

Примечание: маленькая буква и большая буква не идентичны.

Пример:

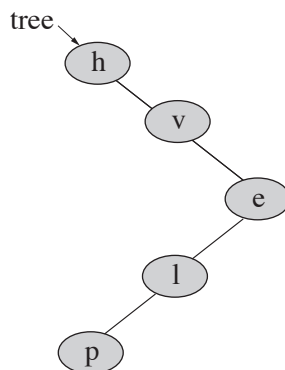
Для данного дерева и строки "help" метод вернет true .



Обратите внимание: продолжение вопроса на следующей странице.

Пример:

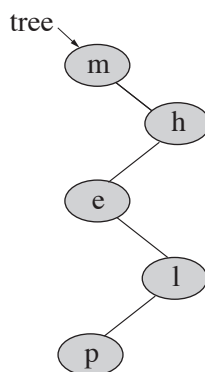
Для данного дерева и строки "help" метод вернет false .



Объяснение: в дереве не существует последовательности знаков, идентичной строке "help".

Пример:

Для данного дерева и строки "help" метод вернет false .



Объяснение: несмотря на то, что в дереве существует последовательность знаков, идентичная строке "help" , метод вернет false , потому что последовательность не начинается в корне дерева.

РАЗДЕЛ ТРЕТИЙ (25 баллов)

В этом разделе вопросы четырёх разных курсов:

- Компьютерные системы и ассемблер [מערכות מחשב ואסמבלי], стр. 16–18.
- Алгоритмы [אלגוריתמים], стр. 19–22.
- Вычислительные модели [מודלים חישוביים], стр. 23
- Объектно-ориентированное программирование [תכנות מונחה עצמים] на языке Java, стр. 24–27;
объектно-ориентированное программирование на языке C#, стр. 28–31.

Ответьте на один вопрос курса, который вы изучали.

Компьютерные системы и ассемблер [מערכות מחשב ואסמבלי]

Если вы изучали этот курс, ответьте на один из вопросов 8–9 (25 баллов).

8. В этом вопросе два пункта (א)–(ב). Между ними нет связи. Ответьте на вопросы обоих пунктов.

(א) Дан фрагмент программы на ассемблере.

```
MOV     SI, 30H
MOV     CL, 4
MOV     CH, 4
MOV     BH, 0
A1: MOV  AX, [SI]
MOV     BL, AL
ROR     BL, CL
DEC     BL
MOV     [SI], BX
ADD     SI, 2
DEC     CH
CMP     CH, 0
JNE     A1
```

Дана карта ячеек памяти от 30H до 36H перед началом выполнения данного фрагмента программы:

address	30H	32H	34H	36H
value	25H	3AH	7BH	93H

Обратите внимание: ячейки памяти имеют размер слова [בגודל מילה].

(1) Проследите с помощью таблицы трассировки [טריס] за выполнением этого фрагмента программы.

Включите в таблицу трассировки столбец для каждого из регистров SI, CL, CH, BL, BH, AX.

Кроме того, напишите соответствующую карту памяти, как она будет выглядеть в конце этого фрагмента программы.

(2) Объясните, что выполняет этот фрагмент программы.

Обратите внимание: продолжение вопроса на следующей странице.

/продолжение на странице 17/

(ב) Перед вами фрагмент программы, написанный на языках Java и C# .

```
if ( x < y + z )
{
    do {
        if ( y >= z || x < y )
            x += y;
        z --;
    } while ( z > 0 );
}
```

Переменные x , y и z – переменные целого типа.

Напишите соответствующий фрагмент программы на ассемблере.

Предположите, что переменные x , y и z являются знаковыми (signed) и хранятся в регистрах AX , BX и CX, соответственно.

Примечание: нет необходимости рассматривать проблему возможного переполнения [גלישה].

9. "Пары с двух сторон" – это пары элементов массива четной длины, соответствующие следующему описанию:
- первая пара состоит из первого элемента и из последнего элемента этого массива,
вторая пара состоит из второго элемента и из предпоследнего элемента этого массива,
третья пара состоит из третьего элемента и из элемента, находящегося перед двумя последними элементами этого массива,
и так далее до последней пары, состоящей из двух элементов, находящихся в середине этого массива.
- "Убывающий четный массив" – массив четной длины, соответствующий следующему условию: суммы "пар с двух сторон" расположены в убывающем порядке, то есть сумма первой пары больше суммы второй пары, сумма второй пары больше суммы третьей пары, и так далее, до последней пары, сумма которой самая маленькая.

Пример убывающего четного массива:

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

Объяснение: массив является "убывающим четным", потому что сумма первой пары (8+5) больше суммы второй пары (6+5), которая больше суммы третьей пары (7+2), которая больше суммы последней пары (1+2).

Пример массива, который не является убывающим четным:

0	1	2	3	4	5
6	8	1	2	3	5

Объяснение: массив не является "убывающим четным", потому что сумма первой пары (6+5) не больше суммы второй пары (8+3), а равна ей.

Во фрагменте данных определены следующие данные:

ARR DB 20 DUP (?)
REZ DB ?

Напишите фрагмент программы, который проверит, является ли массив ARR убывающим четным массивом.

Если да, программа сохраняет число 1 в переменной REZ, иначе она сохраняет в ней число 0.

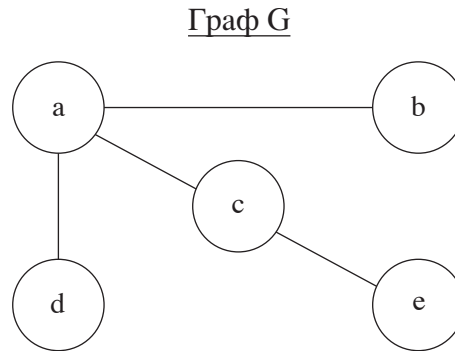
Примечание: нет необходимости проверять, четная ли длина этого массива.

Алгоритмы [אלגוריתמים]

Если вы изучали этот курс, ответьте на один из вопросов 10–11 (25 баллов).

10. В этом вопросе два пункта (א)–(ב). Между ними нет связи. Ответьте на вопросы обоих пунктов.

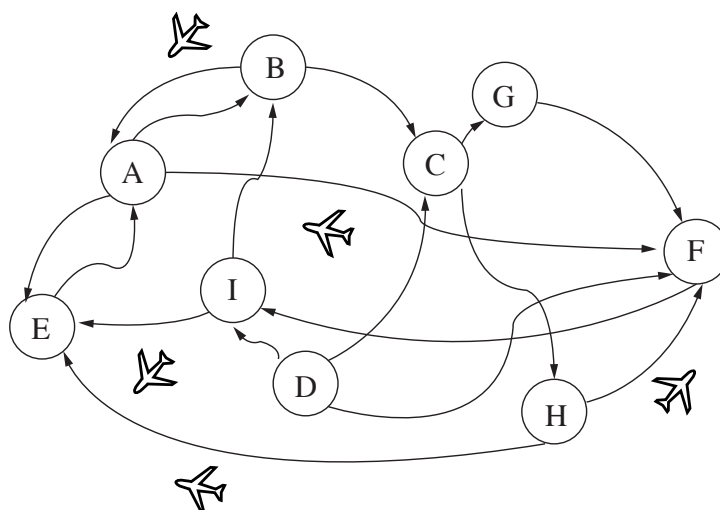
(א) Ниже приведен неориентированный граф $G = (V, E)$.



- (1) Является ли этот граф связанным [קשיר]? Обоснуйте.
- (2) Является ли этот граф двусторонним [דו-צדדי]? Обоснуйте.
- (3) Является ли этот граф полным [מלא (שלם)]? Обоснуйте.
- (4) Является ли этот граф деревом? Обоснуйте.

Обратите внимание: продолжение вопроса на следующей странице.

- (ב) Авиаккомпания обычно осуществляет авиарейсы между странами А , В , С , D , E , F , G , H , I так, как это показано на чертеже ниже:



- (1) Начерите список смежностей, представляющий авиарейсы между этими странами.

Вследствие перемен эта авиакомпания решила отменить несколько авиарейсов в следующих странах:

Страна G – отмена входящих авиарейсов и исходящих авиарейсов

Страна H – отмена рейсов, выходящих в страну F

Страна D – отмена рейсов, выходящих в страну C

Страна A – отмена рейсов, выходящих в страну E

- (2) Начерите список смежностей, который получается после отмен, указанных выше.

- (3) I. Авиталь должна добраться из одной страны в другую.

Для этого она хочет проверить, может ли она добраться в эту страну.

Если может, то она хочет найти какой-либо маршрут перелета в эту страну.

Какой алгоритм она должна использовать для этих целей? Объясните.

Примечание: если есть несколько подходящих алгоритмов, выберите один из них.

- II. Задействуйте алгоритм, о котором вы написали в подпункте (3)I, чтобы найти маршрут перелета из страны A в страну E. Запишите маршрут перелета, который вы нашли.

Необходимо выполнить подробную трассировку [друм] на каждом этапе, в соответствии с алгоритмом, который вы задействовали.

11. В этом вопросе четыре пункта (א)–(ד). Между ними нет связи. Ответьте на вопросы всех пунктов.

(א) Сообщение It's my bank account конвертировано в биты [סיביות], кроме слова my .

Конвертируйте в биты слово my .

Значение Ascii буквы m равно 109 в десятичной системе, а значение Ascii буквы y равно 121 в десятичной системе.

(ב) Перед вами набор битов – m , представляющий значение Ascii слова Hey , и набор битов – key , который является одноразовым блокнотом [פנקס חד-פעמי] (OTP).

m – 010010000110010101111001

key – 110110100100010111101101

Зашифруйте слово Hey посредством метода хог с key .

(ג) Ронит и Юваль зашифровали некоторое сообщение – m , длина которого четное n , посредством метода хог с ключом key и получили зашифрованное сообщение – cipher . Из осторожности они разрезали key пополам. Первую половину ключа key₁ (длиной n/2) Ронит забрала в одно место, а вторую половину ключа key₂ (длиной n/2) Юваль забрал в другое место.

Чтобы расшифровать это сообщение, их подруга Хелит произвела следующие действия по порядку:

- Она взяла у Юваля key₂ и добавила в его начало нули в количестве n/2 (таким образом длина key₂ стала n).
- Она выполнила cipher хог key₂ и получила набор битов. Назовем полученный набор битов halfMessage .
- Она взяла у Ронит key₁ и добавила в его конец число нулей длиной n/2 (таким образом длина key₁ стала n).
- Она выполнила halfMessage хог key₁ и получила набор битов. Назовем полученный набор битов fullMessage .

Напишите, является ли расшифровка корректной (является ли fullMessage сообщением m). Обоснуйте.

Примечание: за ответ без обоснования баллы не начисляются.

Обратите внимание: продолжение вопроса на следующей странице.

(7) Боб и Элис использовали метод CreatePrg/createPrg , чтобы создать ключ PRG , и с его помощью зашифровали сообщение посредством метода хог . Метод получает строку битов [מחרוזת של סיביות] – seed , и длину сообщения, которое нужно зашифровать – len , и возвращает одноразовый блокнот (OTP).

Ниже приведен метод CreatePrg/createPrg, написанный на языках Java и C# .

Ив знает, каков последний бит в строке seed . Может ли она расшифровать это сообщение с одной попытки? Обоснуйте.

Примечание: за ответ без обоснования баллы не начисляются.

На языке C# :	На языке Java :
<pre> public static string CreatePrg (string seed, int len) { string str; if (seed[seed.Length - 1] == '1') str="0"; else str="1"; for (int i = 1; i < len; i ++) { if (str[str.Length - 1] == '0') str += "1"; else str += "0"; } return str; } </pre>	<pre> public static String createPrg (String seed, int len) { String str; if (seed.charAt(seed.length() - 1) == '1') str="0"; else str="1"; for (int i = 1; i < len; i ++) { if (str.charAt(str.length() - 1) == '0') str += "1"; else str += "0"; } return str; } </pre>

Вычислительные модели [מודלים חישוביים]

Если вы изучали этот курс, ответьте на один из вопросов 12–13 (25 баллов).

12. Дан язык L над алфавитом $\{a, b\}$:

$$L = \{a^n b^m \mid n > 0, m \geq 0, m \% 4 = n \% 2\}$$

(*) Ниже приведены 10 слов. Для каждого из них укажите, принадлежит ли это слово языку L , и обоснуйте.

a, ab, aaabbb, aaabbbb, aabb, aaab, aa, bbbb, aaaa, abbb

(*) Постройте полный детерминированный автомат, принимающий язык L .

13. Дан язык L_1 над алфавитом $\{a, b, c\}$:

$$L_1 = \{c^{1+k+n} b^k a^{2n} \mid n, k \geq 1\}$$

(*) (1) Каково самое короткое слово в языке L_1 ?

(2) Постройте стек-автомат, принимающий язык L_1 .

(*) Определим язык L_2 над алфавитом $\{a, b, c, d\}$:

$$L_2 = L_1 \cdot d \cdot R(L_1)$$

Для каждого из приведенных ниже слов укажите, принадлежит ли это слово языку L_2 , и обоснуйте.

ccbaaddaabccc

ccbaadcccbaa

ccbaadaaaabcccc

Примечание: нет необходимости строить автомат для языка L_2 .

Объектно-ориентированное программирование [תכנות מונח בוביי] на языке Java

Если вы изучали этот курс и пишете на языке Java, ответьте на один из вопросов 14–15 (25 баллов).

14. Хозяин магазина "Кольбо" хочет продавать свои товары также по интернету (online).

На первом этапе хозяин магазина решил продавать онлайн только рубашки и брюки.

Для этого были написаны следующие классы:

класс **Clothing** , представляющий одежду

класс **Shirt** , представляющий рубашку

класс **Pants** , представляющий брюки

```
public class Clothing {  
    private int id;           // идентификационный номер товара  
    private String fabric;    // вид материи  
    private String color;     // цвет изделия  
    private double price;     // цена  
}  
  
public class Shirt extends Clothing {  
    private String size;      // размер (S, M, L, XL, XXL)  
}  
  
public class Pants extends Clothing {  
    private int length;       // длина в дюймах  
    private int width;        // ширина в дюймах  
}
```

Считайте, что определены методы `get` и `set` для атрибутов всех классов.

(*) Класс **Cart** представляет корзину покупок. У этого класса три атрибута:

`name` – имя покупателя

`arr` – массив, включающий и рубашки, и брюки

`current` – текущее число товаров в корзине покупок

(1) Напишите заголовок класса `Cart` и его атрибуты.

(2) Начертите схему иерархии, описывающую связь между классами: `Cart` , `Pants` , `Shirts` , `Clothing` .

Обозначьте наследование посредством стрелки  , а включение – посредством символа  .

Обратите внимание: продолжение вопроса на следующей странице.

- (א) На втором этапе хозяин магазина решил продавать онлайн также и книги.

Перед вами класс **Book**, представляющий книгу:

```
public class Book {  
    private int id;           // идентификационный номер товара  
    private String bookName;  // название книги  
    private String author;    // имя автора  
    private double price;     // цена  
}
```

Считайте, что определены методы `get` и `set` для атрибутов класса.

Чтобы массив `arr` в классе `Cart` мог включать также книги, было сделано следующее:

- Определен класс **Product** – товар.
- Тип переменной `arr` в классе `Cart` изменен на `Product`.
- В соответствующих классах (включая класс `Book`) описаны новые отношения наследования между ними и классом `Product`.
- Часть атрибутов, которые были до сих пор в нескольких других классах (включая класс `Book`), перенесены в класс `Product`, в соответствии с принципами объектно-ориентированного программирования (кроме этих атрибутов, в классе `Product` нет других атрибутов)

- (1) Начертите обновленную схему иерархии, описывающую связь между классами:

`Clothing`, `Shirt`, `Pants`, `Book`, `Product`, `Cart`.

Обозначьте наследование посредством стрелки , а включение – посредством символа .

- (2) Напишите заголовок класса `Product` и его атрибуты.
- (3) Напишите заново классы, в которых произошло изменение вследствие добавления класса `Product`.

Обратите внимание: необходимо указать и новые отношения наследования, и изменение, которое произошло в атрибутах класса.

- (א) На последнем этапе хозяин магазина решил объявить о временной распродаже: при покупке онлайн все товары будут продаваться со скидкой 10%, а для покупки одежды (рубашек и брюк) будет дополнительная скидка в 10 шекелей (считайте, что цена остается положительной после скидки).

Дан внутренний метод `[פערלה פנימית]` в классе `Product`:

```
public double getDiscountPrice()
```

Вызов этого метода вернет цену товара (рубашки/брюк/книги) после скидки – соответственно каждому товару.

Реализуйте этот метод в классе `Product` и добавьте метод/ы в соответствующий/ие класс/ы, чтобы выполнялось требуемое действие.

Для каждого добавленного вами метода укажите, к какому классу он добавлен.

Запрещено использовать метод `instanceof` и методы класса `Object` и запрещено изменять атрибуты классов.

/продолжение на странице 26/

15. Даны три класса – Mammal (млекопитающее), Antelope (антилопа) и Beaver (бобр):

```
public class Mammal {
    protected int weight;

    public Mammal (int w) {
        weight = w;
    }
    public int getWeight () {
        return weight;
    }
    public boolean isSame (Mammal other) {
        System.out.println ("In Mammal");
        return (this == other);
    }
}

.....

public class Antelope extends Mammal {
    public Antelope (int w) { super (w); }
    public boolean isSame(Antelope other) {
        System.out.println ("In Antelope");
        return ((other != null) && (this.weight == other.weight));
    }
}

.....

public class Beaver extends Mammal {
    public Beaver (int w) { super (w); }
    public boolean isSame (Mammal other) {
        System.out.println ("In Beaver");
        return ((other != null) && (other instanceof Beaver) && (this.weight == ((Beaver)other).weight));
    }
}

.....
```

Обратите внимание: продолжение вопроса на следующей странице.

Дан класс Program :

```
public class Program {  
    public static void main (String[] args) {  
        Antelope a1 = new Antelope (10);  
        Object a2 = new Antelope (10);  
        Beaver b1 = new Beaver (10);  
        Mammal b2 = new Beaver (10);  
        *****  
    }  
}
```

Класс Program находится в пакете (Package), отличном от класса Mammal .

Подставьте каждую из строк 1–10 , приведенных ниже, в метод main , на место, обозначенное звездочками ***** .

Напишите в своей тетради номер строки и укажите, является ли код корректным или некорректным. Если код корректный – напишите, каков будет вывод [פלט], если код некорректный – объясните почему.

1. System.out.println (a1.weight);
2. System.out.println (((Beaver)a2).getWeight());
3. System.out.println (a1.isSame (a2));
4. System.out.println (a2.isSame (a1));
5. System.out.println (b1.isSame (b2));
6. System.out.println (b2.isSame (b1));
7. System.out.println (a1.isSame ((Beaver)b2));
8. System.out.println (a1.isSame ((Antelope)a2));
9. System.out.println (b1.isSame ((Antelope)a2));
10. System.out.println (b1.isSame ((Beaver)a2));

Объектно-ориентированное программирование [תכנות מונחם] на языке C#

Если вы изучали этот курс и пишете на языке C#, ответьте на один из вопросов 16–17 (25 баллов).

16. Хозяин магазина "Кольбо" хочет продавать свои товары также по интернету (online).

На первом этапе хозяин магазина решил продавать онлайн только рубашки и брюки.

Для этого были написаны следующие классы:

класс **Clothing** , представляющий одежду

класс **Shirt** , представляющий рубашку

класс **Pants** , представляющий брюки

```
public class Clothing {  
    private int id;           // идентификационный номер товара  
    private string fabric;    // вид материи  
    private string color;     // цвет изделия  
    private double price;     // цена  
}  
  
public class Shirt : Clothing {  
    private string size;      // размер (S, M, L, XL, XXL)  
}  
  
public class Pants : Clothing {  
    private int length;       // длина в дюймах  
    private int width;        // ширина в дюймах  
}
```

Считайте, что определены методы Get и Set для атрибутов всех классов.

(8) Класс **Cart** представляет корзину покупок. У этого класса три атрибута:

name – имя покупателя

arr – массив, включающий и рубашки, и брюки

current – текущее число товаров в корзине покупок

(1) Напишите заголовок класса Cart и его атрибуты.

(2) Начертите схему иерархии, описывающую связь между классами: Cart , Pants , Shirts , Clothing .

Обозначьте наследование посредством стрелки  , а включение – посредством символа  .

Обратите внимание: продолжение вопроса на следующей странице.

- (ב) На втором этапе хозяин магазина решил продавать онлайн также и книги.

Перед вами класс **Book** , представляющий книгу:

```
public class Book {  
    private int id;           // идентификационный номер товара  
    private string bookName;  // название книги  
    private string author;    // имя автора  
    private double price;     // цена  
}
```

Считайте, что определены методы `Get` и `Set` для атрибутов класса.

Чтобы массив `arr` в классе `Cart` мог включать также книги, было сделано следующее:

- Определен класс **Product** – товар.
- Тип переменной `arr` в классе `Cart` изменен на `Product`.
- В соответствующих классах (включая класс `Book`) описаны новые отношения наследования между ними и классом `Product`.
- Часть атрибутов, которые были до сих пор в нескольких других классах (включая класс `Book`), перенесены в класс `Product` , в соответствии с принципами объектно-ориентированного программирования (кроме этих атрибутов, в классе `Product` нет других атрибутов)

- (1) Начертите обновленную схему иерархии, описывающую связь между классами:

`Clothing` , `Shirt` , `Pants` , `Book` , `Product` , `Cart` .

Обозначьте наследование посредством стрелки  , а включение – посредством символа  .

- (2) Напишите заголовок класса `Product` и его атрибуты.

- (3) Напишите заново классы, в которых произошло изменение вследствие добавления класса `Product` .

Обратите внимание: необходимо указать и новые отношения наследования, и изменение, которое произошло в атрибутах класса.

- (ג) На последнем этапе хозяин магазина решил объявить о временной распродаже: при покупке онлайн все товары будут продаваться со скидкой 10% , а для покупки одежды (рубашек и брюк) будет дополнительная скидка в 10 шекелей (считайте, что цена остается положительной после скидки).

Дан внутренний метод `[פערלה פנימית]` в классе `Product` :

```
public virtual double GetDiscountPrice()
```

Вызов этого метода вернет цену товара (рубашки/брюк/книги) после скидки – соответственно каждому товару.

Реализуйте этот метод в классе `Product` и добавьте метод/ы в соответствующий/ие класс/ы, чтобы выполнялось требуемое действие.

Для каждого добавленного вами метода укажите, к какому классу он добавлен.

Запрещено использовать методы `is` и `as` и методы класса `Object` и запрещено изменять атрибуты классов.

/продолжение на странице 30/

15. Даны три класса – Mammal (млекопитающее), Antelope (антилопа) и Beaver (бобр):

```
public class Mammal {
    protected int weight;

    public Mammal (int w) {
        weight = w;
    }
    public int GetWeight () {
        return weight;
    }
    public virtual bool IsSame (Mammal other) {
        Console.WriteLine ("In Mammal");
        return (this == other);
    }
}
```

```
.....
public class Antelope : Mammal {
    public Antelope (int w) : base (w) {}
    public bool IsSame(Antelope other) {
        Console.WriteLine ("In Antelope");
        return ((other != null) && (this.weight == other.weight));
    }
}
```

```
.....
public class Beaver : Mammal {
    public Beaver (int w) : base (w) {}
    public override bool IsSame (Mammal other) {
        Console.WriteLine ("In Beaver");
        return ((other != null) && (other is Beaver) && (this.weight == ((Beaver)other).weight));
    }
}
```

Обратите внимание: продолжение вопроса на следующей странице.

Дан класс Program :

```
public class Program {  
    public static void Main (string[] args) {  
        Antelope a1 = new Antelope (10);  
        Object a2 = new Antelope (10);  
        Beaver b1 = new Beaver (10);  
        Mammal b2 = new Beaver (10);  
        *****  
    }  
}
```

Подставьте каждую из строк 1–10 , приведенных ниже, в метод Main , на место, обозначенное звездочками ***** .

Напишите в своей тетради номер строки и укажите, является ли код корректным или некорректным. Если код корректный – напишите, каков будет вывод [פלט], если код некорректный – объясните почему.

1. Console.WriteLine (a1.weight);
2. Console.WriteLine (((Beaver)a2).GetWeight());
3. Console.WriteLine (a1.IsSame (a2));
4. Console.WriteLine (a2.IsSame (a1));
5. Console.WriteLine (b1.IsSame (b2));
6. Console.WriteLine (b2.IsSame (b1));
7. Console.WriteLine (a1.IsSame ((Beaver)b2));
8. Console.WriteLine (a1.IsSame ((Antelope)a2));
9. Console.WriteLine (b1.IsSame ((Antelope)a2));
10. Console.WriteLine (b1.IsSame ((Beaver)a2));

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.