

Задача № 1
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme запишите определение предиката, проверяющего, является ли аргумент символом (литерой) заглавной буквы латинского алфавита. *Не используйте* встроенные предикаты классификации символов.

Утверждена на заседании кафедры ИУ-9

Задача № 2
к экзаменационному билету
по курсу «Основы информатики»

На языке Python (Javascript) запишите определение предиката, проверяющего, является ли аргумент символом (литерой) заглавной буквы латинского алфавита. *Не используйте* встроенные предикаты классификации символов.

Утверждена на заседании кафедры ИУ-9

Задача №3
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme запишите определение процедуры, «удаляющей» из списка повторяющиеся элементы, например:

`(remove-repeats '(1 1 2 4 2)) ⇒ (1 2 4)`

Утверждена на заседании кафедры ИУ-9

Задача №4
к экзаменационному билету
по курсу «Основы информатики»

На языке Python (Javascript) запишите определение функции, «удаляющей» из списка повторяющиеся элементы, например:

`remove_repeats([1, 1, 2, 4, 2]) ⇒ [1, 2, 4]`

Утверждена на заседании кафедры ИУ-9

Задача №5
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme запишите определение процедуры **filter**, принимающей предикат одного аргумента и список и возвращающей список из тех элементов исходного списка, которые удовлетворяют предикату.

Утверждена на заседании кафедры ИУ-9

Задача №6
к экзаменационному билету
по курсу «Основы информатики»

На языке Python (Javascript) запишите свое собственное определение встроенной функции **filter**, принимающей предикат одного аргумента и список и возвращающей список из тех элементов исходного списка, которые удовлетворяют предикату.

Утверждена на заседании кафедры ИУ-9

Задача № 7
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme запишите свое собственное определение встроенной процедуры `map`:

- а) Принимающей процедуру строго одного аргумента и строго один список значений,
- б) Принимающей процедуру произвольного числа аргументов и соответствующее число списков значений.

Утверждена на заседании кафедры ИУ-9

Задача № 8
к экзаменационному билету
по курсу «Основы информатики»

На языке Python (Javascript) запишите свое собственное определение встроенной функции `map`:

- а) Принимающей процедуру строго одного аргумента и строго один список значений,
- б) Принимающей процедуру произвольного числа аргументов и соответствующее число списков значений.

Утверждена на заседании кафедры ИУ-9

Задача №9
к экзаменационному билету
по курсу «Основы информатики»

На языке Python (Javascript) напишите собственное определение функции **reduce**. Через эту функцию определите специальные виды свертки: сумму, произведение, нахождение минимального элемента списка и максимального элемента списка.

Утверждена на заседании кафедры ИУ-9

Задача №10
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme запишите собственные определения встроенных процедур:

- а) `list-ref`,
- б) `length`,
- в) `reverse`.

Утверждена на заседании кафедры ИУ-9

Задача № 11
к экзаменационному билету
по курсу «Основы информатики»

На языке Python (Javascript) запишите определение функции, осуществляющей подсчет элементов списка, удовлетворяющих предикату.

Утверждена на заседании кафедры ИУ-9

Задача № 12
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme запишите определение процедуры, осуществляющей подсчет элементов списка, удовлетворяющих предикату.

Утверждена на заседании кафедры ИУ-9

Задача № 13
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme напишите собственные определения специальных видов свертки: суммы, произведения, нахождения минимального элемента списка и максимального элемента списка.

Утверждена на заседании кафедры ИУ-9

Задача № 14
к экзаменационному билету
по курсу «Основы информатики»

В некоторой программе на языке Python (Javascript) множества представлены в виде списков (Javascript: в виде массивов), элементы которых не повторяются. Используя списковые включения (list comprehensions), запишите определения операций над множествами (Javascript: используйте функции высших порядков для обработки последовательностей):

- а) разности,
- б) симметрической разности,
- в) декартового произведения.

Утверждена на заседании кафедры ИУ-9

Задача № 15
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme запишите определение процедуры, которая принимает выражение вида $a * x + a * y$, записанное на языке Scheme в префиксной нотации (a — целое число, а x и y — переменные или выражения) и возвращает выражение вида $a \cdot (x + y)$, также записанное на языке Scheme в инфиксной нотации и пригодное для вычисления его значения с помощью встроенной процедуры `eval` в окружении, в котором определены соответствующие символы. В случае, если преобразование не может быть выполнена, процедура должна возвращать `#f`. Примеры применения процедуры:

```
(recast '(+ (* 2 x) (* 2 y))) ⇒ (* 2 (+ x y))  
(recast '(+ (* 2 (- a b)) (* 2 (- p q)))) ⇒ (* 2 (+ (- a b) (- p q)))  
(recast '(expt x 2)) ⇒ #f
```

Утверждена на заседании кафедры ИУ-9

Задача № 16
к экзаменационному билету
по курсу «Основы информатики»

Дано равенство вида: $x_1 \star x_2 \star x_3 = x_4 \star x_5 \star x_6$, где $x_1 \dots x_6$ — целые числа в диапазоне от 1 до 100, а \star — операция сложения, деления, вычитания или деления нацело. В одном равенстве могут встречаться как одинаковые, так и разные числа и операции. На языке Python напишите определение функции, которая возвращает все возможные равенства в виде списка строк для каждого переданного ей сочетания чисел слева и справа от знака равенства.

Пример вызова:

```
find([63, 9, 28], [7, 4, 7])
```

Пример результата:

```
63//9*28 = 7*4*7  
63//9+28 = 7*4+7  
63//9+28 = 7+4*7  
63//9-28 = 7-4*7  
63//9//28 = 7//4//7
```

Утверждена на заседании кафедры ИУ-9

Задача № 17
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme напишите определение процедуры для конкатенации символических имен (литеральных констант). Пример применения процедуры:

```
(symbols-append 'foo 'bar) ⇒ foobar  
(symbols-append 'foo '-' 'bar) ⇒ foo-bar
```

Утверждена на заседании кафедры ИУ-9

Задача № 18
к экзаменационному билету
по курсу «Основы информатики»

Используя БНФ, запишите формальную грамматику для логических выражений, записанных в традиционной инфиксной нотации. Грамматика должна учитывать приоритет операторов. В выражениях могут встречаться: константы **True** и **False**, операторы **and**, **or** (бинарные), **not** (унарный), круглые скобки, переменные. Имена переменных могут состоять только из латинских букв. Приоритет операторов убывает в ряду: **not**, **and**, **or**. Порядок вычислений может задаваться круглыми скобками так же, как в арифметических выражениях.

Приведите пример такого выражения и дерево разбора этого выражения.

Утверждена на заседании кафедры ИУ-9

Задача № 19
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme напишите определение процедуры, возвращающей число своих вызовов.

Утверждена на заседании кафедры ИУ-9

Задача № 20
к экзаменационному билету
по курсу «Основы информатики»

На языке Python напишите определение функции, возвращающей число своих вызовов.

Утверждена на заседании кафедры ИУ-9

Задача № 21
к экзаменационному билету
по курсу «Основы информатики»

На языке Python напишите определение класса, реализующего абстрактный тип данных «очередь».

Утверждена на заседании кафедры ИУ-9

Задача № 22
к экзаменационному билету
по курсу «Основы информатики»

На языке Python напишите определение класса, реализующего абстрактный тип данных «стек».

Утверждена на заседании кафедры ИУ-9

Задача № 23
к экзаменационному билету
по курсу «Основы информатики»

На языке Python (Javascript) напишите 2 варианта функции, вычисляющей среднее арифметическое последовательности чисел. В одном варианте используйте императивные управляющие конструкции, во втором – встроенные функции высших порядков.

Утверждена на заседании кафедры ИУ-9

Задача № 24
к экзаменационному билету
по курсу «Основы информатики»

На языке Scheme напишите функцию `drop`, принимающую список и целое число `n` и возвращающую исходный список без `n` первых элементов, например, так:

`(drop '(1 2 3 4) 2) ⇒ (3 4)`

Утверждена на заседании кафедры ИУ-9