## Варианты задач к курсу «Основы алгоритмизации и программирования» групп С-13 и С-21

# Вариант 1

**Задача 1.** Напишите вспомогательную функцию, которая по остатку от деления на 12 введённого пользователем года определяет (и возвращает) название года по китайскому календарю. Соответствие:

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | обезьяны | 6 | тигра |
| 1 | петуха | 7 | кролика |
| 2 | собаки | 8 | дракона |
| 3 | свиньи | 9 | змеи |
| 4 | мыши | 10 | лошади |
| 5 | быка | 11 | овцы |

Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит пользователя ввести год и с помощью функции определяет его название. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** (Числа градины) Для любого натурального числа *n* определено преобразование

|  |  |  |
| --- | --- | --- |
| *F(n)=* | { | *3\*n+1*, если *n* нечетно |
| *n/2*, если *n* четно |

Напишите вспомогательную функцию, которая для заданного числа *n* выполняет преобразование *F* для числа *n* и всех получающихся в результате чисел (градин), пока не получает 1 и возвращает все градины числа n в основную программу. Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит пользователя ввести число *N*, вычисляет и выводит на экран все градины числа *N*. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 3.**  Напишите вспомогательную функцию *f*, которая рисует треугольник по двум сторонам и углу между ними. Напишите вспомогательную функцию *F,* которая по двум сторонам, углу между ними и целому числу *n* рисует *n* равных треугольников с общей вершиной и углом *360/n* между одинаковыми сторонами соседних треугольников. Воспользуйтесь вспомогательной функцией *F* в основной программе, которая в цикле просит пользователя ввести две стороны и угол между ними, числа *n, m* и с помощью функции *F* рисует *n* равных треугольников с общей вершиной, вершины в *m* случайных точках. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет»..

# Вариант 2

**Задача 1.** Напишите вспомогательную функцию, которая по полученным баллам землетрясения по шкале Рихтера печатает описание последствий землетрясения согласно нижеследующей таблице.

|  |  |  |  |
| --- | --- | --- | --- |
| Баллы | Описание | Баллы | Описание |
| >8.0 | Почти все строения рухнули | >4.5 | Повреждены здания, имеющие дефекты. |
| >7.0 | Многие здания разрушены | ≤4.5 | Повреждений практически нет |
| >6.0 | Многие здания значительно повреждены, некоторые разрушены. |  |  |

Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит ввести баллы после чего печатает соответствующее сообщение. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** Напишите вспомогательную функцию, которая в тексте с заданным алфавитом считает количество гласных и согласных букв. Примените функцию в основной программе, которая в цикле просит ввести текстовую строку и печатает количество гласных и согласных букв в строке. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу (все вводимые символы – только из алфавита). Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 3.** Напишите вспомогательную функцию, которая рисует с помощью черепашки эмблему фирмы «Мерседес». Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит пользователя ввести координаты точки и с помощью функции рисует главную диагональ из 10 эмблем «Мерседеса», верхняя эмблема с центром в указанной точке. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 3

**Задача 1.** Напишите вспомогательную функцию, которая переводит температуру в градусах Цельсия в градусы Фаренгейта (*TF*=9/5\**TC*+32). Примените вспомогательную функцию в основной программе, которая в цикле просит ввести температуру в градусах Цельсия и переводит её в градусы Фаренгейта и печатает обе температуры. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** Напишите вспомогательную функцию, которая по данному радиусу вычисляет объём и площадь поверхности шара (*V*=4/3\*π *r*3, *S*=4π\**r*2). Напишите основную программу, которая в цикле просит ввести радиус и выводит на экран монитора с помощью вспомогательной функции объём шара и площадь соответствующего радиусу шара. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Число π импортируйте из библиотеки cmath. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет»..

**Задача 3.** Напишите вспомогательную функцию, которая рисует с помощью черепашки правильный *n*-угольник со стороной данной длины. Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит ввести координаты точки и рисует дополнительную диагональ из 11 правильных *n*-угольников, самый нижний должен быть с вершиной в указанной точке. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 4

**Задача 1.** Напишите вспомогательную функцию, которая по введённому числу *n* определяет совершенное оно или нет. Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит ввести число *N*, определяет и печатает все совершенные числа, меньше или равные числу *N*. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** Напишите вспомогательную функцию, которая заполняет две матрицы *n*×*n* случайными целыми числами, складывает, перемножает матрицы и печатает все четыре матрицы по строкам. Целое число *n* должно быть случайным из отрезка [5, 10]. Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле спрашивает продолжать ли работу и в случае положительного ответа («да») формирует и печатает с помощью вспомогательной функции три матрицы.

**Задача 3.** Напишите вспомогательную функцию, которая с помощью черепашки рисует букву А. Воспользуйтесь вспомогательной функцией в основной программе в цикле просит ввести координаты точки, после чего с помощью функции рисует уголок «юго-запад» из букв А самая высокая точка самой высокой буквы должна совпадать с указанной. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 5

**Задача 1.** Напишите вспомогательную функцию, которая по стороне и прилегающей высоте вычисляет площадь треугольника. Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит ввести основание и прилегающую высоту треугольника, вычисляет площадь и печатает введённые основание, прилегающую высоту и вычисленную площадь треугольника. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** Напишите функцию, которая заполняет две матрицы *m*×*n* случайными целыми числами из интервала [1, 50], складывает матрицы и печатает все три матрицы по строкам. Целые числа *m* и *n* должны быть случайным из отрезка [5, 10]. Воспользуйтесь функцией в основной программе, которая в цикле спрашивает продолжать ли работу и в случае положительного ответа («да») формирует и печатает с помощью функции три матрицы, а в случае отрицательного ответа («нет») завершает работу.

**Задача 3.** Напишите вспомогательную функцию *drawSquares(t, length)*, в которой черепашка *t* рисует два квадрат длины *length:*

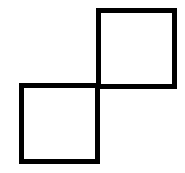


Рис. 1. «Два квадрата»

Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит ввести координаты точки и рисует главную диагональ из 9 рисунков «Два квадрата», самый верхний с вершиной в указанной точке. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 6

**Задача 1.** Напишите вспомогательную функцию вычисления площади круга и длины окружности по данному радиусу. Число π возьмите из библиотеки cmath. Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит ввести радиус, после чего печатает введённый радиус, вычисленные длину окружности и площадь круга. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** Напишите вспомогательную функцию, котораяопределяет является ли данное число *n* простым Воспользуйтесь вспомогательной функцией в основной программе, которая просит ввести число *N* и определяет и печатает все простые числа меньше либо равные *N*. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 3.**  Напишите вспомогательную функцию, которая рисует три вложенных окружности радиусов 10, 20 и 30 пикселов, касающиеся в одной точке (точка касания обязательно сбоку):



Рис. 3. «Три окружности».

Воспользуйтесь вспомогательной функцией в основной программе. Программа должна просить в цикле ввести координаты точки и рисовать из рисунков «Три окружности» уголок «северо-восток». Точка, в которой касаются окружности самого левого рисунка, должна совпадать с «введённой» точкой. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 7

**Задача 1.** Напишите вспомогательную функцию, котораядля данного числа *n* определяет и возвращает в основную программу все его делители *d* (1 < *d* < *n*). Воспользуйтесь вспомогательной функцией в основной программе, которая просит ввести число и печатает все его делители. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** Напишите вспомогательную функцию, которая просит ввести текст с вашим алфавитом и проверяет, является ли текст палиндромом (совпадает с текстом, полученным при записи всех символов в обратном порядке). Примените вспомогательную функцию в основной программе, которая определяет и печатает, является ли введённый пользователем в цикле текст палиндромом. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 3.** Напишите вспомогательную функцию *drawHouse(t, length)*, в которой черепашка *t* рисует дом (квадрат со стороной *length* и равнобедренный прямоугольный треугольник со стороной *length*):

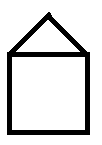


Рис. 2. «Дом»

Воспользуйтесь вспомогательной функцией в основной программе. Программа должна просить в цикле ввести координаты точки и рисовать из рисунков «Дом» уголок «северо-запад», самый нижний домик проходит через указанную точку. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 8

**Задача 1.** Напишите вспомогательную функцию, которая по трём действительным числам вычисляет корни соответствующего квадратного уравнения и печатает их. Если дискриминант <0, то функция должна писать сообщение, что действительных корней нет, если =0, то программа должна сообщить об одном двойном корне. Основная программа в цикле просит ввести три действительных числа, с помощью вспомогательной функции печатает сведения о соответствующих корнях квадратного уравнения. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** Напишите программу, которая генерирует бросание 1-ой, 2-ух, 3-х, 4-х игральных костей. (Пользователь выбирает один из четырёх вариантов, ему выдаётся соответствующий запросу результат). Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет». Генерирование бросаний и обеспечение правильного ввода пользователя оформите в виде вспомогательных функций.

**Задача 3.** Напишите вспомогательную функцию, которая рисует с помощью черепашки эмблему фирмы «Мерседес» с центром в данной точке. Воспользуйтесь функцией в основной программе, которая в цикле просит пользователя ввести целое число *n*, после чего с помощью функции рисует эмблему «Мерседеса», с центром в *n* случайных точках. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 9

**Задача 1.** Напишите вспомогательную функцию, которая переводит температуру по Фаренгейту в температуру по Цельсию и печатает предупреждение «Очень холодно!», если температура меньше -20, печатает «Холодновато», если температура меньше 0, но больше -20, «Не жарко», если температура от 0 до 20, «Сравнительно нормально», -- от 20 до 25, «Жарковато» -- от 25 до 30, и печатает «На улице очень жарко!», если температура больше 30. Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит ввести температуру по Фаренгейту и с помощью функции печатает значение температуры по Цельсию и соответствующее предупреждение. Еще одна вспомогательная функция должна обеспечить правильность ввода данных в программу. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** Программа формирует случайное число в диапазоне [0, 100] и предлагает пользователю отгадать его методом «проб и ошибок». После каждой попытки пользователя программа сообщает меньше загаданное число, чем предположенное пользователем или больше. Процесс отгадывания заканчивается, когда пользователь отгадывает число. В заключительном сообщении программы указывается загаданное число и количество попыток пользователя, затраченное на угадывание. Игровой процесс и обеспечение правильности ввода пользователя оформите в виде вспомогательных функций. Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 3.** Напишите вспомогательную функцию, которая рисует три вложенных окружности радиусов 10, 20 и 30 пикселов, касающиеся в одной точке (точка касания обязательно сбоку):



Рис. 3. «Три окружности».

Воспользуйтесь вспомогательной функцией в основной программе. Основная программа должна просить в цикле ввести координаты точки и рисовать главную диагональ из рисунка «Три окружности», самый верхний рисунок проходит через указанную точку. Обеспечение правильности ввода пользователя оформите в виде вспомогательной функции. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 10

**Задача 1.** Напишите вспомогательную функцию, которая генерирует заданное число *N* пар случайных чисел из интервала [0, 1] и вычисляет приближение для числа π/4. Воспользуйтесь вспомогательной функцией в основной программе, которая просит пользователя задать число *N* и вычисляет приближение для числа π/4, и сравнивает его со значением из библиотеки cmath. Обеспечение правильности ввода пользователя оформите в виде вспомогательной функции. Основная программа должна работать пока пользователь не введёт вместо *N* отрицательное число.

**Задача 2.** Напишите программу, которая просит пользователя ввести текст с вашим алфавитом, после чего печатает текст без гласных букв и без гласных букв. Формирование строк для печати и обеспечение правильности ввода пользователя оформите в виде вспомогательных функций. Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 3.** Напишите вспомогательную функцию, которая рисует с помощью черепашки эмблему фирмы «Мерседес» с центром в данной точке. Еще одна вспомогательная функция должна обеспечивать правильность ввода данных пользователем. Воспользуйтесь вспомогательными функциями в программе, которая просит пользователя ввести целое число *n*, после чего рисует эмблему «Мерседеса» с центром в *n* случайных точках. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 11

**Задача 1.** «Московская кофейня на паях» продаёт кофе по цене 1230 руб. за кг плюс стоимость пересылки. За пересылку установлена цена 100 руб. +50 руб. за каждый килограмм кофе. Есть ограничение – не более 3 кг кофе в одном заказе. Напишите программу вычисления стоимости заказа. Программа должна проверять правильность ввода (добиваться ввода натурального или положительного числа с плавающей точкой). Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 2.** Напишите вспомогательную функцию, которая просит пользователя ввести фамилию, имя и год рождения. После чего приветствует пользователя и сообщает ему сколько лет он прожил. Примените вспомогательную функцию в основной программе, которая в цикле просит ввести фамилию, имя и год рождения, с помощью функции приветствует пользователя и сообщает ему сколько лет он прожил. Обеспечение правильности ввода пользователя оформите в виде вспомогательной функции (функций). Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 3.** Напишите вспомогательную функцию, которая рисует с помощью черепашки *n* радиальных отрезков из общего центра, соседние отрезки образуют угол *360/n*. Воспользуйтесь вспомогательной функцией в программе, которая просит пользователя ввести два целых числа *n* и *m*, после чего рисует фигуры из *n* радиальных отрезков с центром в *m* случайных точках. Обеспечение правильности ввода пользователя оформите в виде вспомогательной функции. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 12

**Задача 1.** Напишите вспомогательную функцию, которая проверяет, является ли заданный текст палиндромом. Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит ввести текстовую строку и проверяет является ли текстовая строка палиндромом. Обеспечение правильности ввода пользователя оформите в виде вспомогательной функции (текст должен состоять только из символов вашего алфавита). Основная программа должна работать до тех пор, пока пользователь откажется вводить очередную строку текста. (На вопрос программы «Продолжить? д/н» ответит «н»)

**Задача 2.** Две точки на плоскости задаются своими координатами (х1, y1) и (x2, y2). Напишите программу вычисления и печати наклона отрезка, соединяющего две точки, введённых пользователем: . Программа должна проверять правильность ввода (добиваться ввода целых чисел или чисел с плавающей точкой и не вертикального наклона). Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 3.** Напишите вспомогательную функцию *drawRectangle(t, side1, side2),* которая рисует прямоугольник со сторонами *side1* и *side2.* Далее напишите основную программу, которая просит ввести координаты точки А, длину и ширину прямоугольника и рисует квадрат из прямоугольников, левый нижний прямоугольник проходит через точку А, сторона квадрата – 7 прямоугольников. Обеспечение правильности ввода пользователя оформите в виде вспомогательной функции. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 13

**Задача 1.** Напишите программу, которая координатам двух введённых пользователем точек сопоставляет и печатает расстояние между ними: . Программа должна проверять правильность ввода (добиваться ввода целых чисел или чисел с плавающей точкой). Вычисление расстояния (без печати) оформите в виде вспомогательной функции. Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 2.** Напишите вспомогательную функцию, которая в тексте с вашим алфавитом считает количество гласных и согласных букв. Примените функцию в основной программе, которая в цикле просит ввести текстовую строку и печатает количество гласных и согласных букв в строке. Программа должна проверять правильность ввода пользователя (все введённые символы только из алфавита). Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 3.** Напишите вспомогательную функцию *drawSquare(t, Length)*, в которой черепашка *t* рисует квадрат длины *Length*. Далее напишите основную программу, которая просит ввести координаты точки А, длину квадрата и рисует прямоугольник 7×11, состоящий из квадратов, левый нижний квадрат проходит через точку А. Еще одна вспомогательная функция должна проверять правильность ввода (добиваться ввода целых чисел). Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 14

**Задача 1.** Напишите вспомогательную функцию для вычисления площади треугольника по формуле Герона. Пользователем задаются три стороны треугольника *a, b, c*. Площадь вычисляется следующим образом: , . Еще одна вспомогательная функция должна проверять правильность ввода (добиваться ввода натуральных или положительных чисел с плавающей точкой и соблюдения неравенств треугольника). Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 2.** Напишите вспомогательную функцию, котораяопределяет является ли данное число *n* простым Воспользуйтесь вспомогательной функцией в основной программе, которая просит ввести число *N* и определяет и печатает все простые числа меньше либо равные *N*. Еще одна вспомогательная функция должна проверять правильность ввода (добиваться ввода натуральных чисел больших 1. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 3.** Напишите вспомогательную функцию, которая рисует с помощью черепашки солнце с *n* лучами.

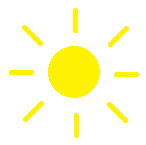


Рис. 4. «Солнце с 8 лучами».

Воспользуйтесь вспомогательной функцией в основной программе, которая просит пользователя ввести два целых числа *n* и *m*, после чего рисует рисунок «Солнце с *n* лучами» в *m* случайных точках. Еще одна вспомогательная функция должна проверять правильность ввода. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 15

**Задача 1.** Напишите вспомогательную функцию, которая вычисляет длину лестницы, наклонённую к дому, необходимую для достижения заданной высоты. Высота (height) и угол наклона лестницы (angle) являются входными данными программы. Для вычисления длины (length) воспользуйтесь формулой: . Еще одна вспомогательная функция должна обеспечитьть правильность ввода (добиваться ввода положительных чисел). Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 2.** Напишите вспомогательную функцию, которая по данным, определяющим два круга на плоскости (круг на плоскости определяется тремя числами, первые два числа координаты центра, третье – радиус) определяет, пересекаются круги или нет. Воспользуйтесь вспомогательной функцией в основной программе. Основная программа должна в цикле просить ввести данные о двух кругах и сообщать пересекаются круги или нет. Еще одна вспомогательная функция должна обеспечитьть правильность ввода данных пользователем. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 3.** Напишите вспомогательную функцию *drawRectangle(t, side1, side2),* которая рисует прямоугольник со сторонами *side1* и *side2.* Далее напишите основную программу, которая в цикле просит ввести координаты точки А, длину и ширину прямоугольника и рисует из прямоугольников крест, прямоугольник, стоящий на пересечении вертикальной и горизонтальной составляющей креста, должен проходить через точку А. Еще одна вспомогательная функция должна обеспечить правильность ввода данных пользователем. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 16

**Задача 1.** Напишите вспомогательную функцию, которая по трём введённым сторонам треугольника определяет, существует ли треугольник с такими сторонами или нет. Далее напишите основную программу, которая в цикле просит ввести длину трёх отрезков и с помощью вспомогательной функции определяет существует такой треугольник или нет. Еще одна вспомогательная функция должна обеспечить правильность ввода данных пользователем. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** Нумерологи претендуют на то, что способны определить персональные черты, основываясь только на «численном значении» имени. Численное значение имени получается с помощью суммирования численных значений всех букв фамилии (имени, отчества), «а» заменяется на 1, «б» на 2, «в» на 3 и т.д. Напишите программу, которая вычисляет численное значение введённого набора слов. Вычисление численных значений текста и обеспечение правильности ввода текста оформите в виде вспомогательных функций. Программа должна добиваться ввода только слов. Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 3.** Напишите вспомогательную функцию, которая рисует с помощью черепашки правильный *n*-угольник со стороной данной длины. Воспользуйтесь вспомогательной функцией в основной программе, которая в цикле просит ввести длину стороны, количество сторон и координаты вершины, после чего с помощью функции рисует из правильных *n*-угольников уголок «юго-восток», самый высокий многоугольник с вершиной в указанной точке. Еще одна вспомогательная функция должна обеспечить правильность ввода данных пользователем. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 17

**Задача 1.** Штраф за превышение скорости в Урюпинске 1500 руб. плюс 150 руб. за каждые 1.5 км/час сверх нормы плюс штраф 6000 руб. за скорость свыше 150 км/час. Напишите вспомогательную функцию, которая получает норму скорости и фактическую скорость и выводит величину штрафа или пишет что скорость в пределах дорожных правил. Еще одна вспомогательная функция должна обеспечить правильность ввода данных пользователем. Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 2.** Напишите вспомогательную функцию, которая определяет, за какое время удваивается вклад для данных годовых процентов. Ввод это годовые проценты, вывод это число лет, за которые удваивается вклад. Еще одна вспомогательная функция должна проверять правильность ввода пользователя. Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 3.** Напишите вспомогательную функцию, которая рисует с помощью черепашки эмблему фирмы «Мерседес» с центром в данной точке. Воспользуйтесь функцией в программе, которая просит пользователя ввести координаты точки, после чего с помощью функции рисует из эмблем «Мерседеса» пересекающиеся диагонали. Еще одна вспомогательная функция должна проверять правильность ввода пользователя. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 18

**Задача 1.** Гипотеза Гольдбаха утверждает, что каждое четное число является суммой двух простых чисел. Напишите программу, которая получает положительные числа от пользователя, проверяет, что они четные и находит для них все пары простых положительных чисел, которые в сумме дают само число. Проверку правильности ввода и нахождение всех пар простых чисел оформите в виде вспомогательных функций. Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 2.** Наибольший общий делитель (НОД) двух целых чисел можно вычислить с помощью алгоритма Евклида. Начиная со значений m и n мы последовательно применяем формулу n, m = m, n%m, пока m не станет равным 0. В этот момент *n* = НОД. Напишите вспомогательные функции, которые вычисляют НОД двух чисел с помощью описанного алгоритма и проверяют правильность ввода (добиваться ввода цифр). Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 3.**  Напишите вспомогательную функцию, которая рисует с помощью черепашки солнце с *n* лучами.

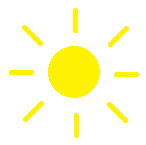


Рис. 4. «Солнце с 8 лучами».

Воспользуйтесь вспомогательной функцией в основной программе, которая просит пользователя ввести натуральные числа *m* и *n*, после чего рисует *m* рисунков «Солнце с *n* лучами» в виде главной диагонали из *m* рисунков. Еще одна вспомогательная функция должна проверять правильность ввода. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 19

**Задача 1.** Напишите вспомогательную функцию, которая для введённой строки текста выдаёт строку всех букв в алфавитном порядке. Основная программа просит ввести текст и выдаёт введённый текст и буквы текста в алфавитном порядке. Еще одна вспомогательная функция должна проверять правильность ввода. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

**Задача 2.** Напишите вспомогательную функцию, которая получает число секунд с начала дня, а возвращает число часов, минут и секунд. Еще одна вспомогательная функция должна проверять правильность ввода (не только чисел, но и то, что заданное время не выходит за пределы суток). Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 3.** Напишите вспомогательную функцию *drawSquare(t, Length)*, в которой черепашка *t* рисует квадрат длины *Length*. Далее напишите основную программу, которая в цикле просит ввести координаты точки А, длину квадрата и рисует дополнительную диагональ из квадратов, самый нижний квадрат должен проходить через точку А. Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».

# Вариант 20

**Задача 1.** Напишите вспомогательную функцию, которая получает число дюймов, а возвращает число ярдов, футов и дюймов, Еще одна вспомогательная функция должна проверять правильность ввода пользователя. Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 2.** Программа просит ввести натуральное число *n*. Генерирует массив длины *n* со случайными целыми числами из интервала [-*n, n*]. Генерация массива и определение сумма каких элементов больше (положительных или отрицательных) оформите в виде вспомогательной функции. Еще одна вспомогательная функция должна проверять правильность ввода *n* (добиваться ввода цифр). Программа должна работать до тех пор, пока на вопрос: «Продолжить? (да/нет): » последует ответ «нет».

**Задача 3.** Напишите вспомогательную функцию, которая рисует с помощью черепашки *n* радиальных отрезков из общего центра, соседние отрезки образуют угол *360/n*. Воспользуйтесь вспомогательной функцией в основной программе, которая просит пользователя ввести два целых числа *n* и *m*, после чего рисует фигуры из радиальных отрезков с центром в *m* случайных точках. Еще одна вспомогательная функция должна проверять правильность ввода *m* и *n* (добиваться ввода цифр). Основная программа должна работать до тех пор, пока пользователь на вопрос «Продолжить? (да/нет): » ответит «нет».