



**Добро пожаловать в  
Компьютерную Академию ШАГ**

# КОМПЬЮТЕРНАЯ АКАДЕМИЯ ШАГ ГОРОД

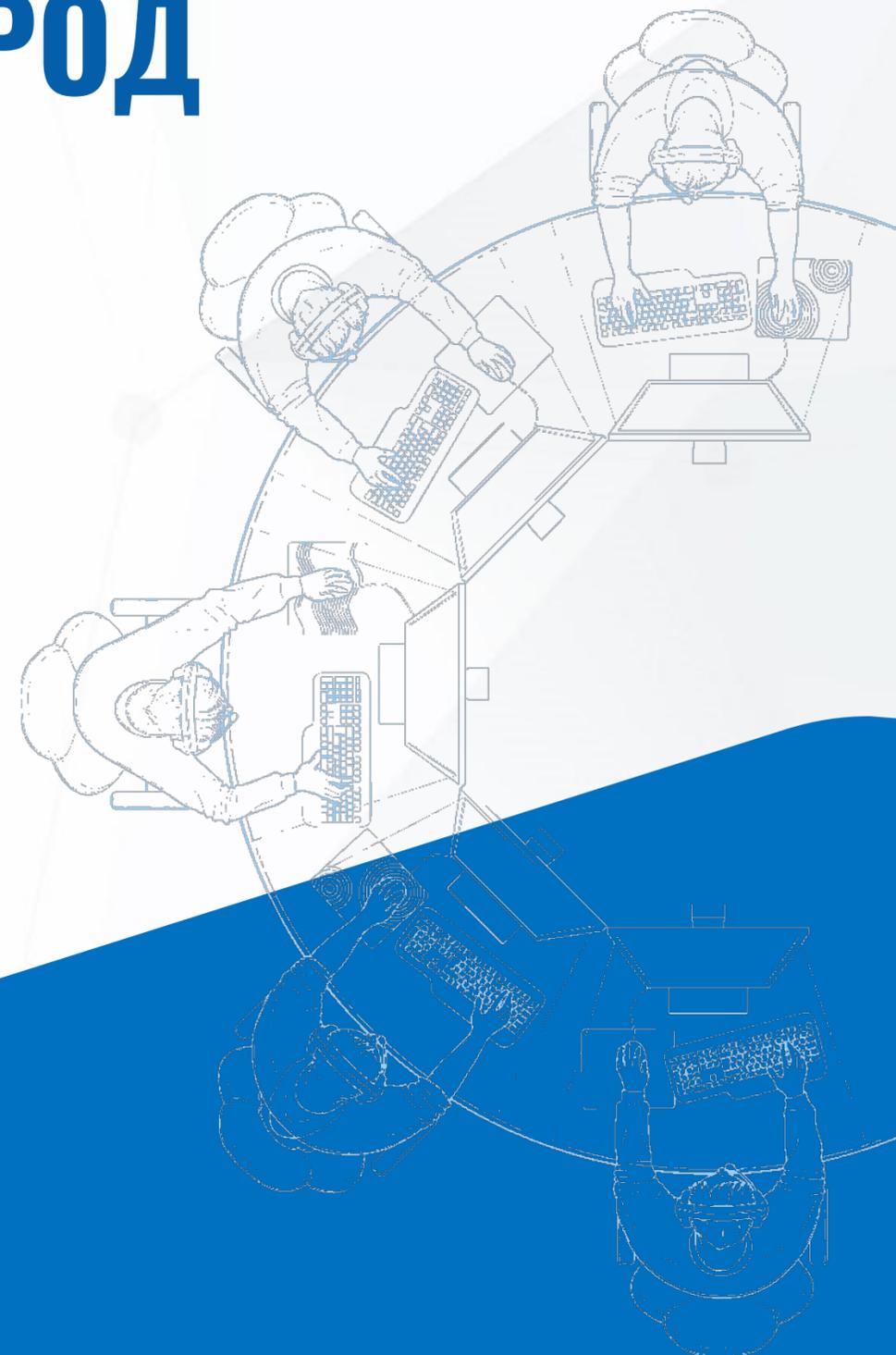
Директор филиала – Регина  
Тимофеева

Менеджер учебного процесса –  
Екатерина Таушканова, Лилия  
Муратора

Преподаватели:

Баторин Николай(ПШ 14)

Баторин Николай(ПШ16)



# РАСПИСАНИЕ ЗАНЯТИЙ

Первый Шаг

**Длительность: 2 года**

**Номер группы: ПШ15, ПШ20**

**Первое занятие:**

ПШ15 (Воскресенье): 16:30 – 17:50

ПШ 20 (Суббота): 16:30 – 17:50



# КАНИКУЛЫ



**Зима:**  
С 30.12 по 9.01



**Весна:**  
С 1 – 3.05, с  
8-9.05



**Лето:**  
с 31.05.2021



# 1 ГОД = 72 УЧ. ЧАСА

## Виды платежей:

**По частям**  
**9 платежей**

**За Семестр**  
**2 платежа**

**За Год**  
**1 платеж**

**+1 страховой/обеспечительный**

При отсутствии оплаты по графику, страховой платеж используется, как основной.  
При отсутствии основного и страхового платежей студент отстраняется от занятий.

# СМЕНА ФОРМЫ ОПЛАТЫ

Написать заявление на смену формы оплаты **МОЖНО ТОЛЬКО В** определенные периоды:

- В начале учебного года - на любую форму
- В конце семестра только на "Семестровую"

Смена формы возможна только 1 раз за учебный год.

```
type
TCustomer : {customer_id : integer,
              customer_name : string,
              active : bool,
              salary : integer};

var
customer : TCustomer;
customers_data : array of TCustomer;
i : integer;
upload_buffer : mainCore->Buffer;

function UpdateCustomerData(customer_id : integer; new_data : TCustomer)
{
    customers_temporary_data = GetCustomersData(all_active_customers_data);
    with customers_temporary_data do
        Sort(MinToMax, 0, customers_temporary_data.length);
        customer_records = mainCore->Modify(customers_temporary_data, customer_id, new_data);
        virtualized_customer_data = mainCore->Virtualize(customers_temporary_data);
    }

    for (i = 0 to virtualized_customer_data.length)
    {
        if (virtualized_customer_data[i] instance_of mainCore->TCustomer)
        {
            virtualized_customer_data[i, 0] = mainCore->Evaluate(customer_id, new_data);
            virtualized_customer_data[i, 1] = mainCore->Evaluate(customer_id, new_data);
        }
    }

    customer = mainCore->GetInput();

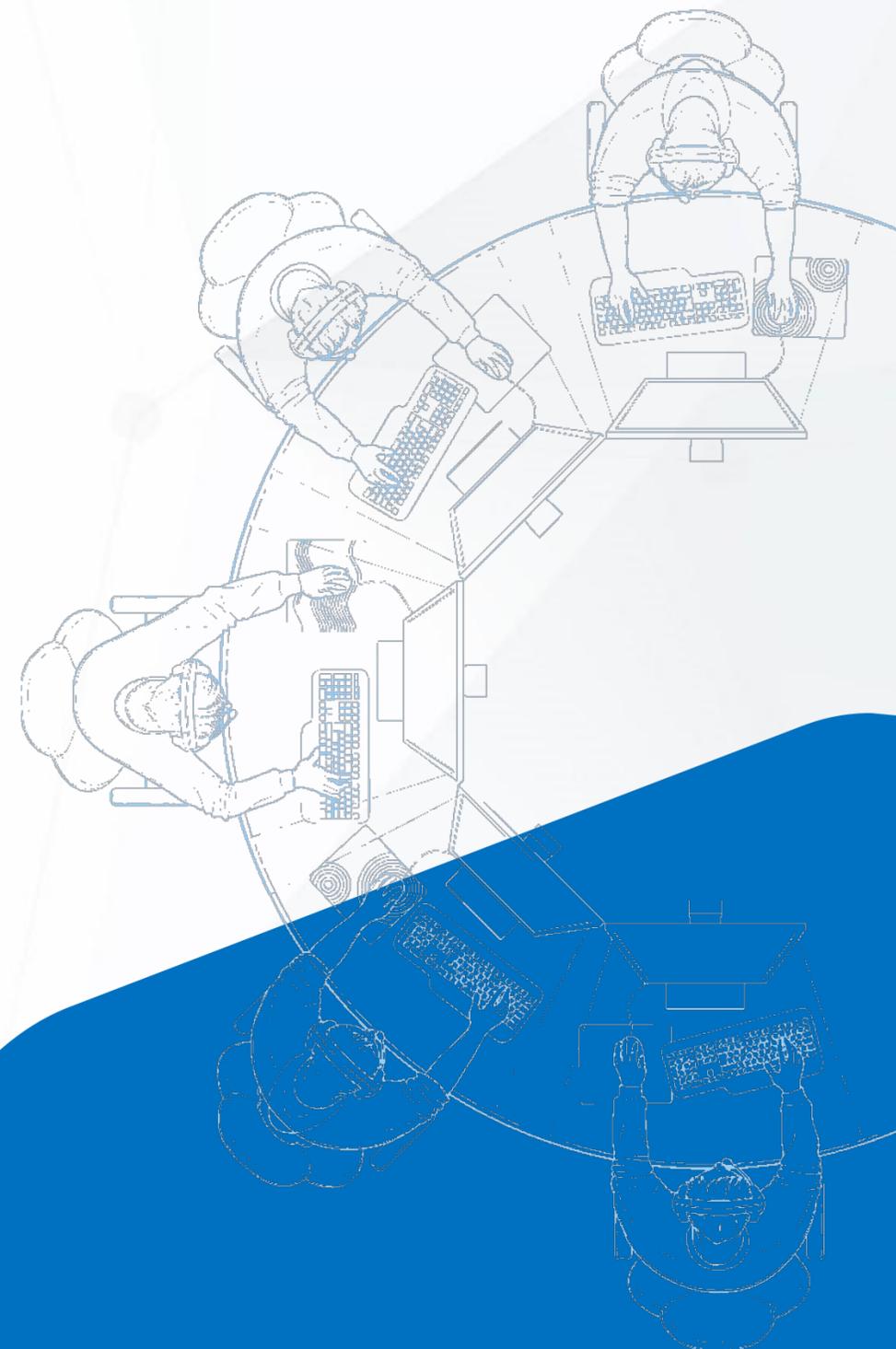
    upload_buffer->initialize();
    if (upload_buffer <> 0)
    {
        upload_buffer->data = UpdateCustomerData(id; customer);
        upload_buffer->state = transmission;
        mainCore->SendToVirtualMemory(upload_buffer);
        mainCore->SendToProcessingCenter(upload_buffer);
    }
}
```

# ПРОПУСК ЗАНЯТИЙ

Если вы понимаете, что вынуждены пропустить занятие, обязательно сообщайте об этом в Viber учебной части.

## Отработка пропусков:

- Отработка по конспектам с возможностью связи с преподавателем;
- Отработка в параллельной группе;
- Есть возможность организовать индивидуальную консультацию (платно).



# ЧТО МЫ БУДЕМ ИЗУЧАТЬ

- IT Start
- Робототехника
- Создание видео - YouTube
- Мультипликация
- Моделирование 3D мира
- Введение в программирование на Scratch

```
type
TCustomer : {customer_id : integer,
              customer_name : string,
              active : bool,
              salary : integer};

var
customer : TCustomer;
customers_data : array of TCustomer;
i : integer;
upload_buffer : mainCore->Buffer;

function UpdateCustomerData(customer_id : integer; new_data : TCustomer)
{
    customers_temporary_data = GetCustomersData(all_active_customers_data);
    with customers_temporary_data do
    {
        Sort(MinToMax, 0, customers_temporary_data.length);
        customer_records = mainCore->Modify(customers_temporary_data, customer_id, new_data);
        virtualized_customer_data = mainCore->Virtualize(customers_temporary_data);

        for (i = 0 to virtualized_customer_data.length)
        {
            if (virtualized_customer_data[i] instance_of mainCore->globalCustomer)
            {
                virtualized_customer_data[i, 0] = mainCore->Evaluate(salary, customer_records);
                virtualized_customer_data[i, 1] = mainCore->Evaluate(customer_name, customer_records);
            }
        }
    }
}

customer = mainCore->GetInput();

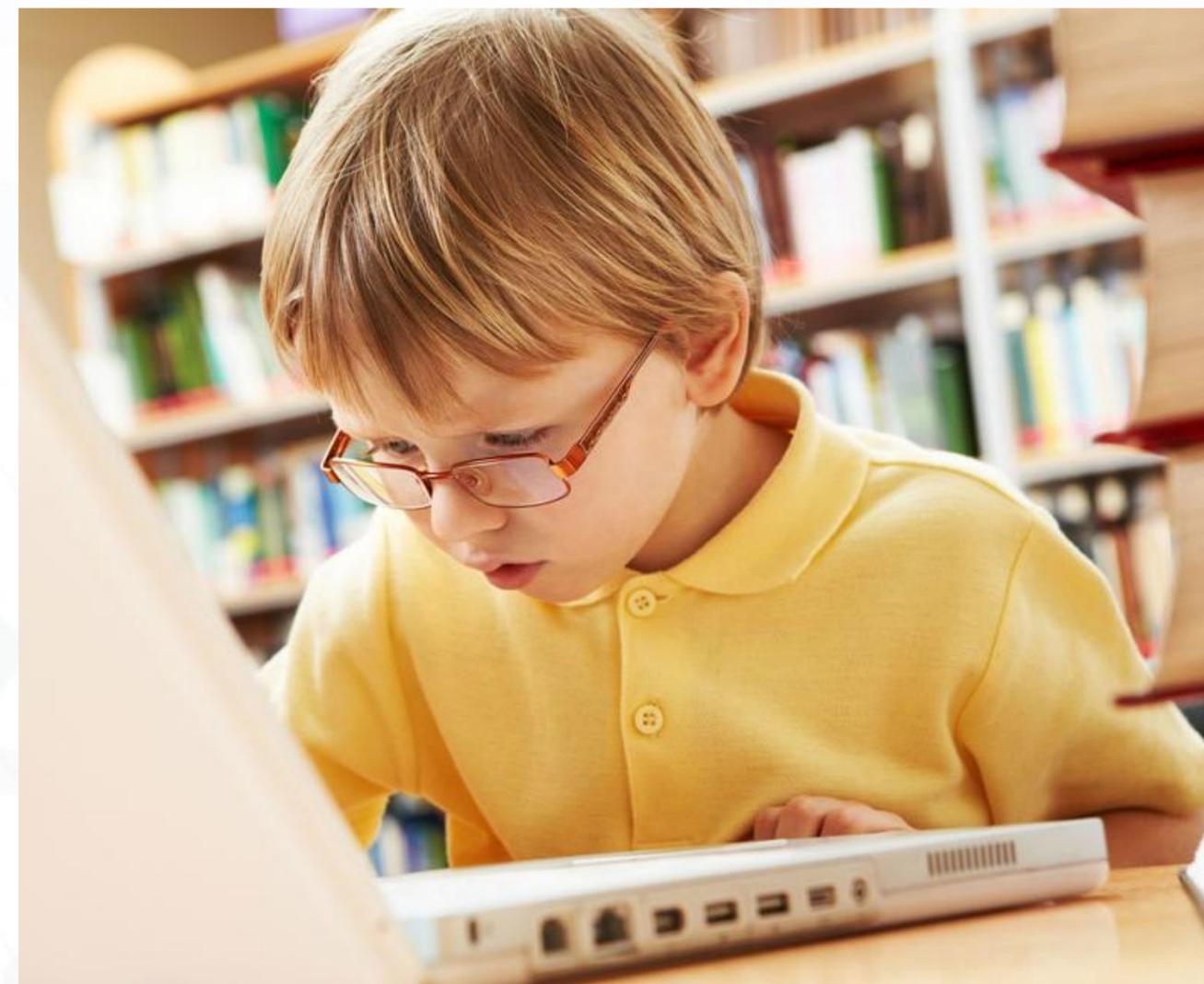
upload_buffer->initialize();
if (upload_buffer <> 0)
{
    upload_buffer->data = UpdateCustomerData(id; customer);
    upload_buffer->state = transmission;
    mainCore->SendToVirtualMemory(upload_buffer);
    mainCore->SendToProcessingCenter(upload_buffer);
}
}
```

## Описание

Знакомство с возможностями операционной системы Windows, создание файлов и папок, работа с изображениями, презентациями, текстовыми документами, аудио и видеофайлами. Обучение быстрой печати.

## Этапы курса:

- Аппаратное обеспечение персонального компьютера.
- Программное обеспечение персонального компьютера. Основы работы с ОС Windows 8/10.
- Обучение быстрой печати
- Операции с файлами. Флешка.
- Поиск и архивирование файлов и антивирусные программы.
- Интернет и веб-браузеры.
- Электронная почта. Облачные хранилища.
- Редактирование текста и таблиц в Google Docs.



IT START



## Робототехника

### Описание

Развитие навыков командной работы и креативности через конструирование и программирование роботов LEGO WeDo для выполнения различных задач, которые можно задать алгоритмами.

### Этапы курса:

- Увлечь студентов изучением технического дизайна и конструирования роботов;
- Научить конструировать и программировать роботов LEGO WeDo 2.0;
- Научить применять датчики и создавать модели роботов в зависимости от поставленной задачи.

## Описание

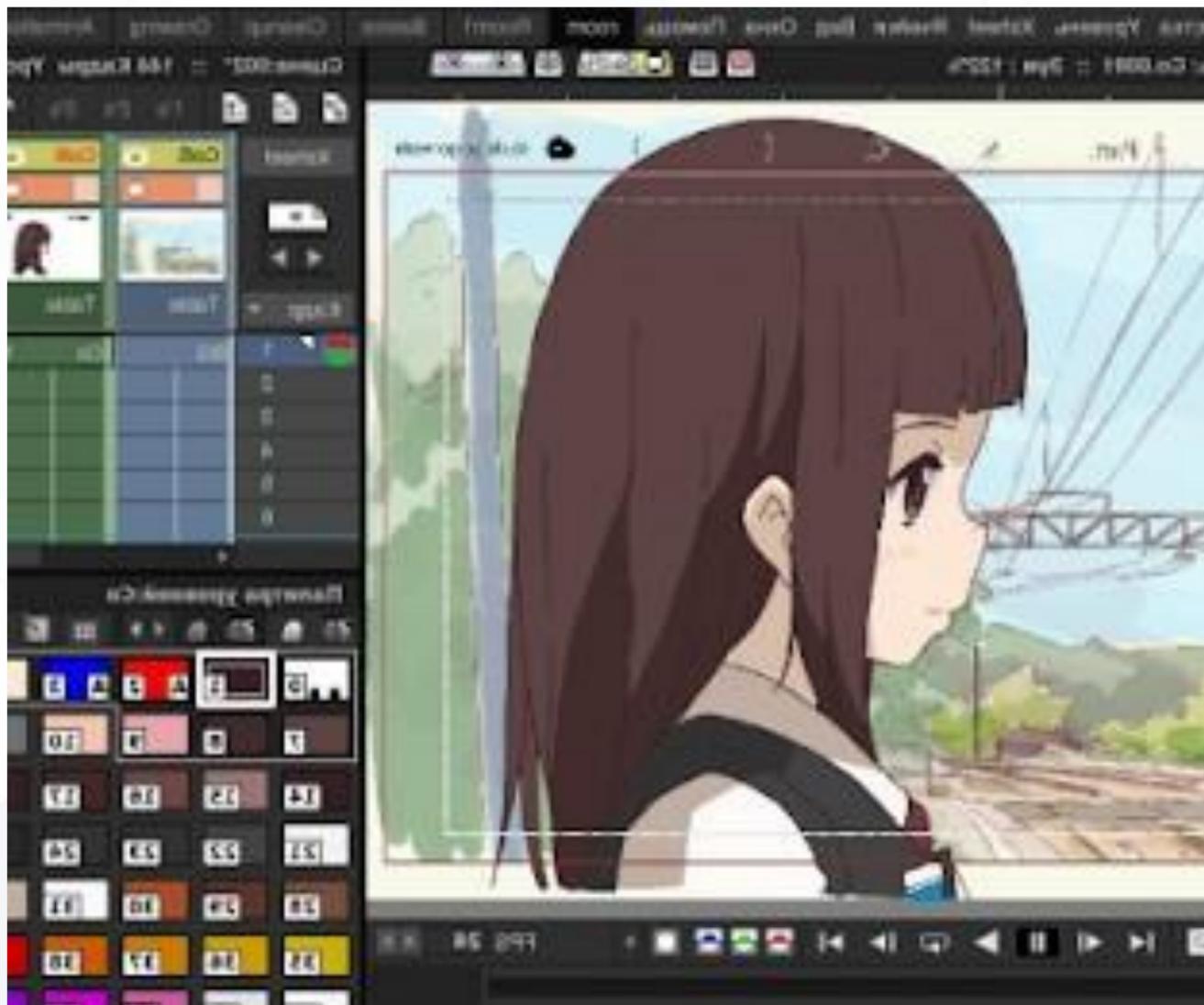
Основы съемки на профессиональный фотоаппарат и телефон. Создание видеоистории и презентации с фотографиями группы. Простая обработка фотографий в Paint.net. Создание простых эффектов на фотографиях. Изучение правил композиции. Практические занятия по разным стилям съемки. Ведение блогов и каналов Youtube. Съёмка и обработка видео стоп-моушн видео в Adobe Sparg, наложение звуковой дорожки и переходов между фрагментами видео, создание слайд-шоу из изображений, добавление субтитров.

## Этапы курса:

- Портретная фотография. Стикеры
- Предметная съемка
- Наложение эффекта
- Создание видеоистории
- Регистрация и создание канала
- Создание ролика в стиле топ-3 самых известных



Создание видео - YouTube



## Мультипликация

### Результат курса

Создание изображений в программе Paint. Net. Коррекция изображений, использование слоев, режимов наложение, редактирование уровней, применение визуальных эффектов и градиентных заливок, работа со шрифтами.

### Этапы курса:

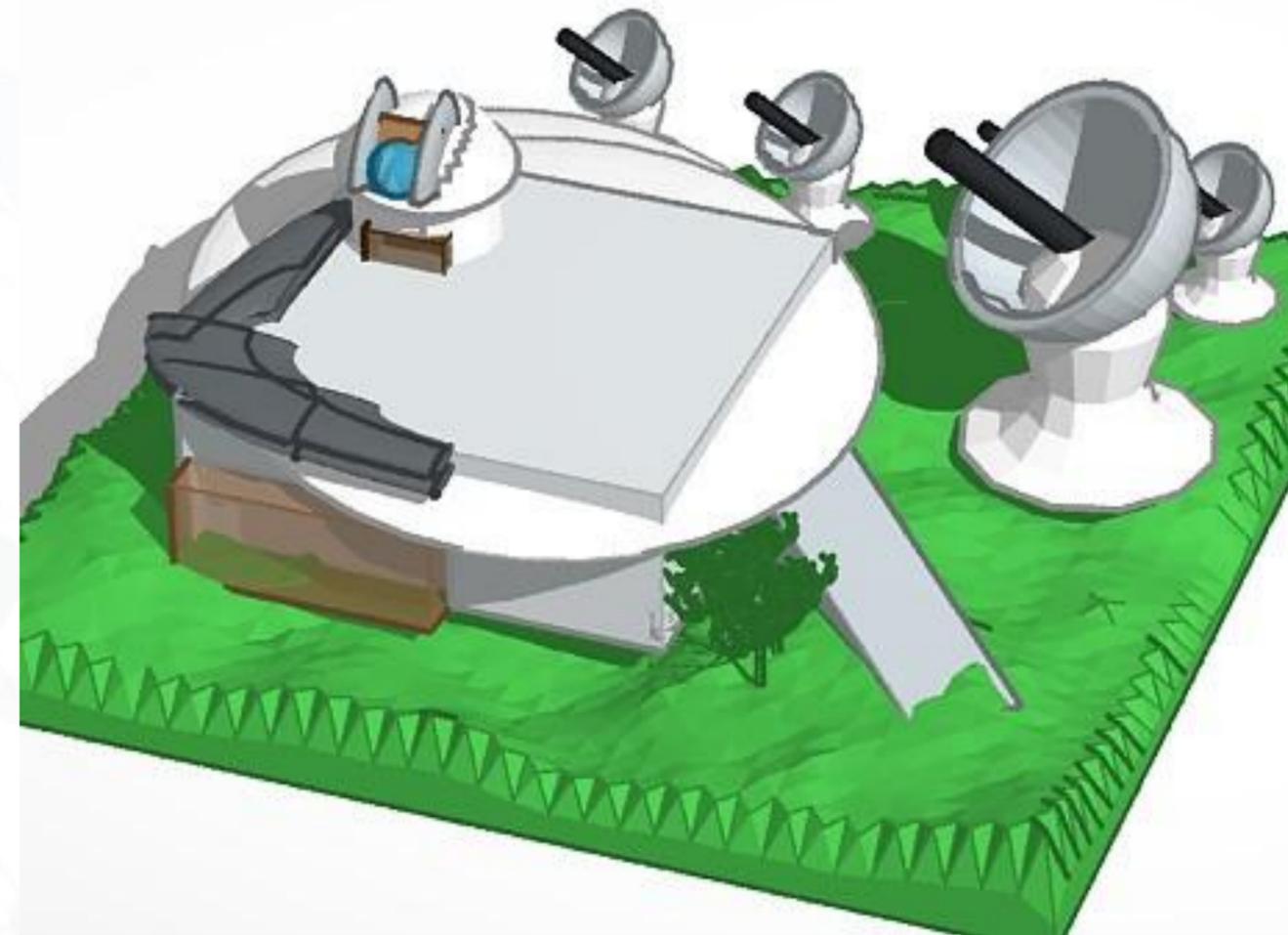
- Цифровой рисунок. Рисуем галактику.
- Использование цвета
- Разделение фотографии на части
- Создание главного героя
- Анимация
- Создание анимации город будущего

## Результат курса

Изучение инженерного дизайна через создание трехмерных моделей. Создание моделей машин, самолета, дома, замка, корабля, космодрома, города.

### Этапы курса:

- Введение в моделирование.
- Моделирование транспорта
- Моделирование космодрома
- Проектирование железной дороги
- Моделирование лунного здания.
- Моделирование Lego робота
- Моделирование города



Моделирование 3D мира

## Введение в программирование на Scratch



## Результат курса

Изучение основ программирования при помощи визуального редактора кода Scratch3. Программирование движения персонажей, их анимация, создание случайных событий, работа со звуком. Создание простых 2D-игр.

## Этапы курса:

- Создание диалога между двумя друзьями с добавлением звуковых эффектов. Создание последовательности блоков ожидания.
- Создание случайных аудио истории. Изменение звуков и добавление разных инструментов и звуков.
- Программирование приложения для создания рисунков при помощи стрелок и специальных клавиш для выбора цветов и ширины линии.
- Программирование конструктор Lego в Scratch. Использование системы координат. Добавление строительных блоков с разными размерами и цветами.



# MyStat - электронный дневник студента

Web-версия [mystat.itstep.org](http://mystat.itstep.org)



GET IT ON  
**Google Play**



Download on the  
**App Store**

# ЧТО ПОНАДОБИТСЯ НА ЗАНЯТИЯХ

- Сменная обувь;
- Ручка, тетрадка - по желанию.

## На первое занятие:

- Соглашение на то, что ребёнок может сам уходить домой.
- Соглашение на размещение фото учебного процесса в соцсетях.

Соглашения нужно отдать МУП или оставить на ресепшене.



# ПРАВИЛА ПОВЕДЕНИЯ НА ЗАНЯТИЯХ

1. Приходим на занятия за 10-15 мин до начала урока.
2. Не разрешается употреблять напитки и еду во время занятий.
3. Не разрешается пользоваться телефонами и планшетами, если это не предусмотрено заданием от учителя.
4. Если что-то не понятно или что-то пропустили, смело говорите об этом преподавателю.
5. При обнаружении любых проблем с компьютером или MyStat, важно сразу сказать об этом преподавателю или сообщать в учебную часть.
6. Мы уважаем себя и коллег, поэтому на занятиях не мешаем друг другу и поддерживаем.
7. Бережно относимся к технике и остальному имуществу. Заказчик обучения несет финансовую ответственность.

# КОНТАКТЫ

**Почта: [zavuch\\_chel@itstep.org](mailto:zavuch_chel@itstep.org)**

**Номер телефона: +7 (351) 220-20-41**

**Чат в Viber: +7 (999) 333-79-64**

```
type
TCustomer : {customer_id : integer,
              customer_name : string,
              active : bool,
              salary : integer};

var
customer : TCustomer;
customers_data : array of TCustomer;
i : integer;
upload_buffer : mainCore->Buffer;

function UpdateCustomerData(customer_id : integer; new_data : TCustomer)
{
    customers_temporary_data = GetCustomersData(all_active_customers_data);
    with customers_temporary_data do
    {
        Sort(MinToMax, 0, customers_temporary_data.length);
        customer_records = mainCore->Modify(customers_temporary_data, customer_id, new_data);
        virtualized_customer_data = mainCore->Virtualize(customer_records);
    }
    for (i = 0 to virtualized_customer_data.length)
    {
        if (virtualized_customer_data[i] instance_of mainCore->TCustomer)
        {
            virtualized_customer_data[i, 0] = mainCore->Evaluate(customer_id);
            virtualized_customer_data[i, 1] = mainCore->Evaluate(new_data);
        }
    }
}

customer = mainCore->GetInput();

upload_buffer->initialize();
if (upload_buffer <> 0)
{
    upload_buffer->data = UpdateCustomerData(id; customer);
    upload_buffer->state = transmission;
    mainCore->SendToVirtualMemory(upload_buffer);
    mainCore->SendToProcessingCenter(upload_buffer);
}
```



**Спасибо за внимание!**  
**Желаем успехов в учебе!**