

## БЛОКОВО ПРОГРАМИРАНЕ ЗА УЧИТЕЛИ И УЧЕНИЦИ

Людмила Атанасова Дерменджиева<sup>1</sup>, Румяна Йорданова Папанчева<sup>2</sup>

<sup>1</sup>Първо основно училище „Св. Св. Кирил и Методий“, гр. Гоце Делчев

<sup>2</sup>Университет „Проф. д-р Асен Златаров“, гр. Бургас

## BLOCK PROGRAMMING FOR TEACHERS AND STUDENTS

Lyudmila Dermendzhieva<sup>1</sup>, Romyana Papancheva<sup>2</sup>

<sup>1</sup>First Primary School „St. Cyril and St. Methodius“, Goce Delchev

<sup>2</sup>University „Prof. Dr A. Zlatarov“, Burgas

**Abstract.** The paper presents some concrete examples from the author's experience in teaching Scratch to primary school students. The environment Scratch is described. Some example-block codes are presented. Conclusions concerning programming skills at primary school are formulated.

**Keywords:** Scratch, block programming, programming, early age, digital skills

Технологиите, информацията, динамиката на обществения и икономическия живот, определят нуждата от адекватно образование, което да изгради грамотни по критериите на съвременното им личности. Грамотността днес е на качествено различно ниво от грамотността на предходните поколения. Грамотният днес човек трябва да чете и пише, както и да работи с информация като я използва, анализира, съхранява, предава и т.н. В глобален план образователните системи се опитват да отговорят на нуждата да се формират знания, умения, компетенции, отношения, чрез които гражданите на обществото да намерят своето качествено и достойно място като лична и професионална реализация и стандарт на живот.

В отговор на тези съвременни нужди и очаквания са и нормативните изменения, настъпили у нас с приемането на нов закон за предучилищно и училищно образование<sup>1</sup> и последвалите го документи. В наредба 5/30.11.2015 г. за общообразователната подготовка на министъра на образованието и науката се определят общообразователните учебни предмети, чрез които ще се осъществява общообразователната подготовка на учениците в Република България (Чл. 3)<sup>2</sup>. Общообразователните предмети са 20 и сред тях е новият учебен предмет „Компютърно моделиране“. В наредбата е записано, че общообразователните предмети са свързани с ключовите компетентности, които трябва да се формират у обучаемите. Учебните предмети информатика, информационни технологии и компютърно моделиране са основополагащи за придобиване на дигитална компетентност.

Една от областите на компетентност при изучаване на Компютърно моделиране е областта „Алгоритми“. В резултат от обучението ученикът трябва да:

- Познава една визуална среда за блокново програмиране, с която може да създаде дигитално съдържание.
- Използва визуална среда за реализация на линеен алгоритъм.
- Използва изучаваната визуална среда за експериментиране.
- Създава проект, включващ повторения и условия.
- Създава проект, който включва анимация с повече от един обект и използване на звук и текст.
- Представя в реална и виртуална среда свой проект.

Според НАРЕДБА № 12 от 01.09.2016 г. за статута и професионалното развитие на учителите, директорите и другите педагогически специалисти<sup>3</sup> учителят по Компютърно моделиране трябва има присъдена професионална квалификация „Начален учител“ или да е учител по информатика или информационни технологии с допълнителна квалификация „Начален учител“. Това изискване поставя сериозни предизвикателства към педагогическата колегия в началното училище. Преподаването на компютърно моделиране изисква базови познания от областта на информатиката и информационните технологии. Учителите

трябва първо да бъдат квалифицирани, за да могат да преподават качествено теми, свързани с дигитални устройства, дигитална идентичност, алгоритми, базови умения за програмиране.

Първият випуск, който ще изучава Компютърно моделиране в трети клас е випуск 2016. Само след една учебна година предстои третокласниците да се запознаят с учебното съдържание на новия учебен предмет. „Акцентът в обучението в трети клас е върху усвояване на знания и умения за работа с дигитални устройства, работа с файлове, създаване на анимирани проекти с използване на алгоритми с условия и повторения чрез визуална среда за блоково програмиране. Реализирането на компютърните модели във визуална среда се подготвя с нагледни материали в позната за учениците среда и изпълнение на алгоритми със средства на тази среда – албуми с блокове и пъзели, лесни за ръчно управление роботизирани устройства и др.“<sup>4</sup>

За формиране на алгоритмично мислене и основни знания по програмиране се използват визуални среди за програмиране, при които детето вижда веднага резултата от своята програма. Една от най-широко използваните среди за визуално програмиране с устойчиво развитие от повече от 10 години е платформата Scratch – разработка на Масачузетски технологичен институт (MIT)<sup>5</sup>. SCRATCH е среда за програмиране на интерактивни истории, игри и анимации и онлайн общност за споделяне с хора от всички краища на света. Scratch е проектирана да се използва според сложността на разглежданите задачи от ученици от III до VIII клас.

Scratch предлага блокова среда за програмиране. Интерфейсът на средата е лесен и интуитивен за работа. Scratch е достъпен на повече от 40 езика и се използва в повече от 150 различни страни. С помощта на цветни блокове, обучаемите „строят“ своите програми. Всяка група команди е с определена цветова идентификация, което ориентира още по-добре ученикът коя група команди трябва да използва. Блоковите команди се подреждат чрез действие влачене и пускане. Избегнато е писането на код и акцентът е поставен върху формирането на логическо и алгоритмично мислене.

Scratch позволява работа както с начинаещи, така и с напреднали. Със средствата на средата могат да се създадат малки анимирани, интерактивни проекти с проста логика. Средата поддържа взаимодействие и с хардуерни елементи. В Scratch 2.0 е въведен и Extension Protocol, който позволява взаимодействие на софтуера с периферия – микрофон, камера, джойстик, LEGO, платки ... Това позволява да се създадат и комплексни проекти със сложна логика и ефектни крайни продукти.

Изследователите на MIT насочват вниманието си и към формиране на алгоритмично и логическо мислене и в предучилищна възраст. За да го направят достъпен и за по-малки деца, пет водещи организации /National Science Foundation, LEGO Foundation, British Telecom, Adobe Systems и Code-to-Learn Foundation/ адаптират интерфейса и езика към възможностите на децата между 5 и 7 години. Така се появява ScratchJr.

Scratch може да се използва за различни образователни и развлекателни цели – математически и научни проекти, анимирани истории, интерактивно изкуство и музика. Scratch проектите могат лесно да бъдат качени на сайта на Scratch. След като проектът е на сайта, всеки може да го изпълни в браузъра си, да го коментира или да го изтегли, за да разглежда и изучава създадените кодове.

В помощ на читателя, който не познава средата е разработено Приложение 1 към настоящата статия. В него е дадена информация за достъп до средата, възможност на инсталация на редактор и кратко описание на интерфейса на софтуерната платформа.

На база многогодишен опит на авторите от работа със средата Scratch с ученици и проведени десетки обучение с ученици и учители на национално ниво, авторите предлагат кратки методически насоки за постепенно усвояване и надграждане на уменията за работа със средата Scratch за начинаещи. Методическите насоки са актуални както учители, които се запознават със средата, така при работата на учителя с ученици за формиране на базови знания и умения по програмиране.

Етапи на запознаване със средата за визуално програмиране и формиране на дигитални умения за създаване на компютърна програма:

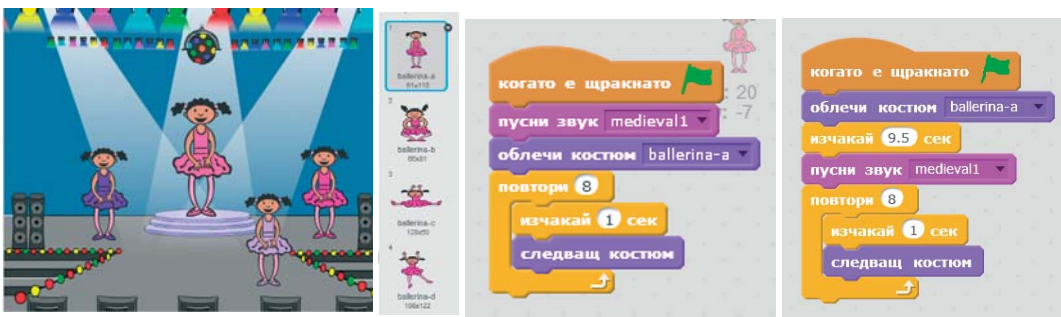
1. Запознаване със средата и основните ѝ елементи, обекти, функции. (Прил. 1).
2. Създаване на слайд-шоу чрез смяна на декора. Добавяне на музикален фон. Запознаване с цикличен оператор – „повтаряй винаги“.
3. Създаване на анимация чрез:
  - Смяна на костюмите (Пример 1).
  - Смяна на цвета.
  - Дорисуване.

- Движение на обекта.
4. Работа с текст. Управление на времето. Създаване на комикс (Фигура 2).
  5. Управление на събития:
    - Работа с условни оператори.
    - Работа със сензори (сетива) – допир до герой, до цвят, разстояние до герой (Пример 2).
    - Контрол на събития – натиснат клавиш, избран герой.
  6. Работа с координатната ос.
  7. Чертане на фигури (Пример 3)
  8. Аритметични операции. Работа с променливи.
  9. Комуникация между обектите. Работа със съобщения.
  10. Разработване на комбинирани проекти (Пример 4, 5):
  11. Разработване на приложение, управляващи хардурени компоненти. Работа с уеб камерата на компютъра.

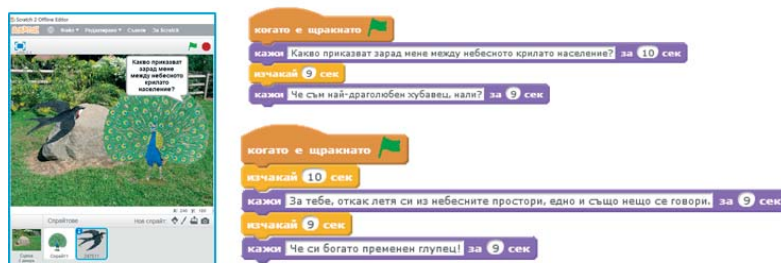
Към описаните етапи за запознаване със средата Scratch и за формиране на начални знания и умения за програмиран даваме няколко примера, илюстриращи избрани акценти.

Пример 1. Да се създаде проект „Балерини“. Учителят – балерина изпълнява 8 такта музикален етюд. След което трите балерини-ученици го поварят заедно с учителя.

Работи се със спрайт от галерията. Балерината има четири костюма. Скриптът от Фигура 1 показва възможни програми, които да накарат балерината-учител и балерините-ученици да танцуват под съпровода на музика от галерията на програмата. Външният вид на героите се управлява чрез промяна на размера и редактиране на костюмите, чрез ново оцветяване с помощта на вградения редактор. При създаване на кода на една от балерините-ученици, останалите могат да се получат чрез копиране и допълнително редактиране на външния вид.



Фигура 1. Анимации чрез смяна на костюми



Фигура 2. Комикс по „Паун и лястовица“ – Стоян Михайловски

При създаването на комикс трябва да се прецени и контролира времето, необходимо за визуализиране на репликите на героите.

Пример 2. Да се създаде анимация, при която приказан герой показва буквата на първия звук от названието на посочен предмет.

Задание – когато мишката е над даден предмет да се показва буквата на първия звук от названието на предмета. Когато не е над избран предмет да се появява приказния герой.

Конкретна реализация – създаване на подходяща сцена и поставяне върху нея на предметите – картина, маса и стол. При избор на масата да се появи буква М, при избор на картината – да се появи буква К, при избор на стола – да се появи буква С, иначе – да се появи вълшебна фея. Изборът става чрез задържане на мишката над съответния предмет (фигура 3).

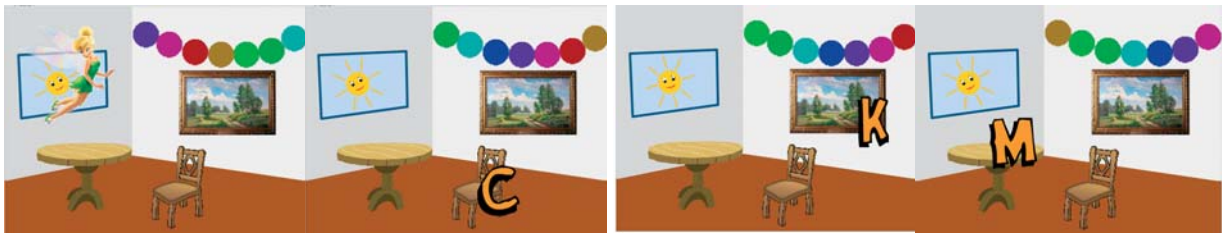
Алгоритъм за решаване на задачата.

1. Разработване и подготовка на дизайна на проекта.

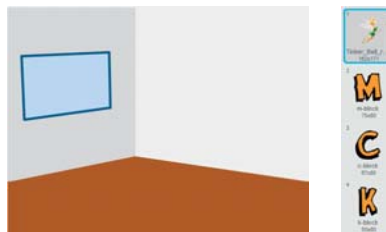
- С помощта на графичен редактор (MS Paint, Paint.Net или др.) се създава сцената на проекта с един декор (Фигура 4).
- Създават се 3 спрайта – стол, картина и маса, от изображения.
- Създава се спрайтът – основен герой, на който се задават 4 костюма (Фигура 4). Костюмите се добавят от галерията на програмата Scratch.
- Вмъкват се няколко спрайта – цветни кръгове за украса на сцената.

2. Програмиране.

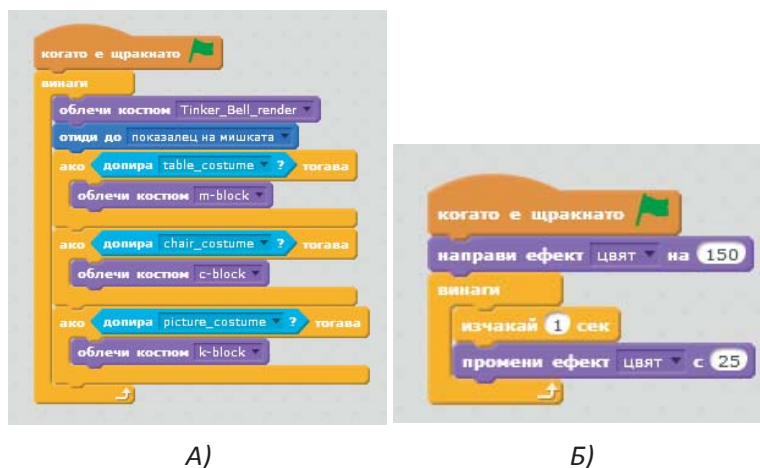
- Логиката, която следва програмата е следната – първоначално спрайтът има външен вид фея и отива до показалецът на мишката. Така при движение на мишката се движи и спрайтът. Използват се условни оператори и сензори. Ако основният спрайт се докосне до картината, външният му вид се изменя в буква К и т.н. На Фигура 5а е даден примерен скрипт за спрайта.
- Украсата на стаята е реализирана от цветни кръгове, които циклично променят цвета си. Използва се ефект – промяна на цвят. Първоначално се задават различни изходни цветове на различните кръгове. Примерен скрипт на спрайт-цветен кръг е даден на фигура 5б.
- Стартирането и спирането на изпълнението на програмата става чрез класическите за средата Scratch инструменти – зеления флаг за старт и червения кръг за стоп.



Фигура 3. Основен спрайт – 4 костюма – фея, буква С, буква К, буква М



Фигура 4. Декор и основен спрайт



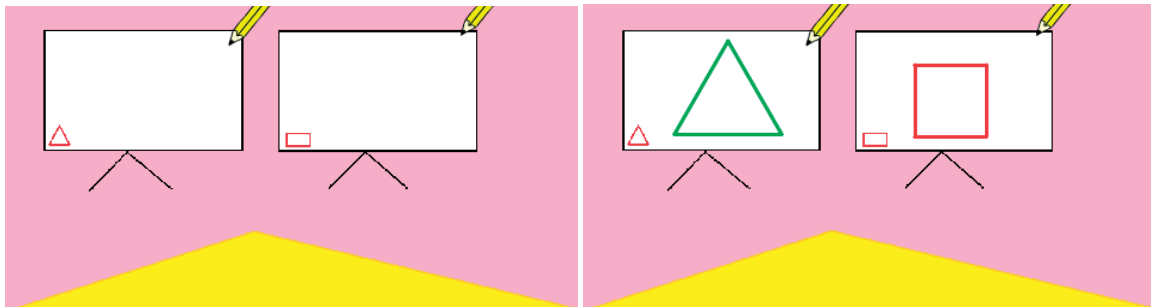
Фигура 5. Скриптове

Пример 3. Да се онагледят изчертаването на квадрат и на триъгълник.

Задание – въвеждат се два молива. При щракване върху единия – се изчертава квадрат. При щракване върху другия – се изчертава равностранен триъгълник.

Конкретна реализация – създаване на подходяща сцена и поставяне върху нея на двата молива. При

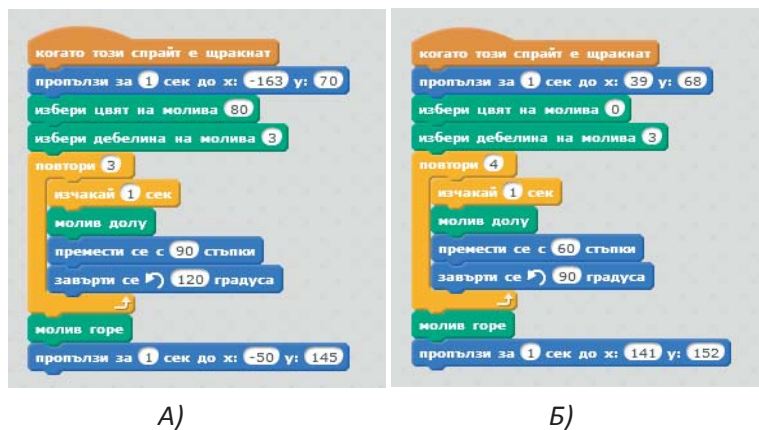
щракване върху единия – се изчертава квадрат. При щракване върху другия – се изчертава равностранен триъгълник (фигура 6).



Фигура 6. Чертане на триъгълник и квадрат

Алгоритъм за решаване на задачата.

1. Разработване и подготовка на дизайна на проекта.
  - С помощта на графичен редактор (MS Paint, Pain.Net или др.) се създава сцената на проекта с един декор.
  - Създават се 2 спрайта – двата молива от галерията на програмата.
2. Програмиране.
  - Логиката, която следва програмата за изчертаване на триъгълник е следната: моливът се позиционира в изходно положение. Избира се цвят и дебелина на линията. За изчертаване на триъгълник се използва цикъл с 3 повторения – преместване по дължината на страната на триъгълника и завъртане на 120 градуса. (Фигура 7). След завършване на цикъла моливът се повдига, за да не оставя следа и се връща обратно в изходна позиция.
  - Логиката, която следва програмата за изчертаване на квадрат е следната: моливът се позиционира в изходно положение. Избира се цвят и дебелина на линията. За изчертаване на квадрат се използва цикъл с 4 повторения – преместване по дължината на страната на квадрата и завъртане на 90 градуса. (Фигура 7).
  - Стартирането и спирането на изпълнението на програмата става чрез щракване върху обект – единия или другия молив.

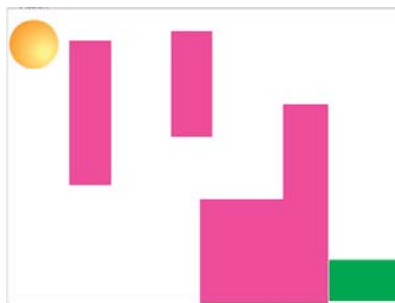


Фигура 7. Скриптове

Пример 4. Разработване на компютърна игра „Лабиринт“.

Задание. Да се разработи компютърна игра, при която обект трябва да бъде преведен през лабиринт, като за движение на обекта се използват стрелките за навигация – напред-назад, наляво-надясно. Да има критерий за край. Да има действие при грешка.

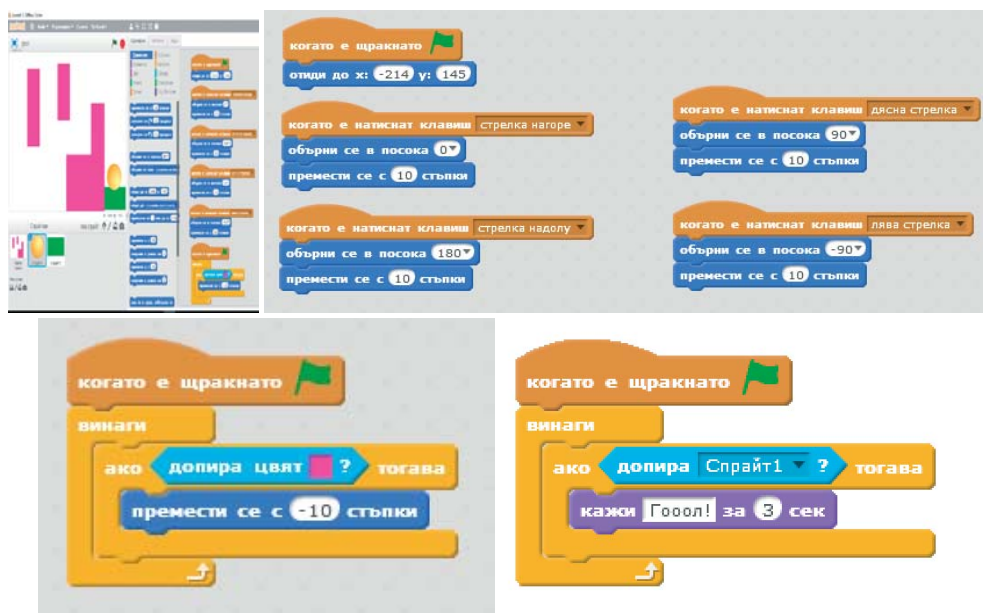
Конкретна реализация. Очертаване на лабиринт с правоъгълник коридори с указано начало и край. Обектът е топка, която трябва да се премести от началото до края на лабиринта. При докосване до стена, топката отскача. (Като вариант на играта топката може да се връща обратно в начална позиция). Краят на пътя е означен със зелен правоъгълник. Играта приключва при достигане на топката до зеления правоъгълник.



Фигура 8. Игрално поле

Алгоритъм за решаване на задачата.

1. Разработване и подготовка на дизайна на проекта (фигура 8).
  - С помощта на редактора Paint Editor или друг графичен редактор (MS Paint, Paint.Net) се създава сцената на проекта с един декор.
  - Създават се 2 спрайта: топката – от галерията на програмата и зеления правоъгълник – създаден с редактора на програмата Scratch – Paint Editor.
2. Програмиране.
  - Логиката, която следва програмата е следната: при стартиране на играта топката заема първоначално зададените ѝ координати.
  - Когато е натиснат клавиш стрелка, топката се обръща в указаната посока и прави 10 стъпки. Ако топката срещне препятствие, тя отскача назад по пътя си с 10 стъпки. При достигане на топката до зеления правоъгълник играта свършва и се появява съобщение „Гол“ (фигура 9).



Фигура 9. Игра „Лабиринт“

Пример 5. Разработване на компютърна игра „Бягай от акулата“.

Задание. Да се разработи компютърна игра, при която акулата иска да изяде малка рибка. Рибката трябва да избяга от акулата. Движението на рибката е чрез клавишите-стрелки за навигация – напред-назад, наляво-надясно. Да има критерий за край на играта, зависещ от точкова система.

Конкретна реализация. За фон – изображение на морско дъно. Обектите са акулата и рибата. Акулата се движи самостоятелно по случайна траектория. Целта е рибата да се измества така, че да се избягва акулата. Управлението на движението на рибата е чрез стрелките за навигация. При среща на акулата с рибата се натрупва 1 точка. Играе се за определено време. Целта е играчът да има възможно по-малко на брой точки.

Алгоритъм за решаване на задачата.

1. Разработване и подготовка на дизайна на проекта (фигура 11).
  - Сцената се избира от галерията на програмата.

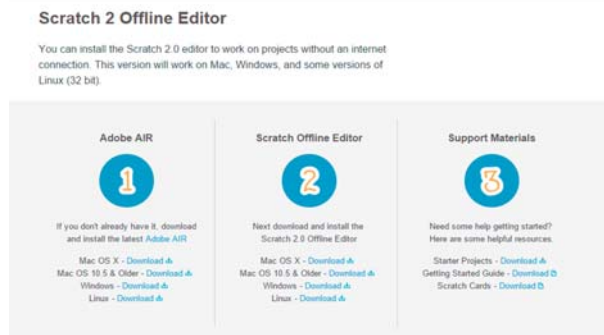
- Създават се 2 спрайта – акула и малка рибка – от галерията на програмата.
2. Програмиране.
- Логиката, която следва програмата е следната: при стартиране на играта броячът на точки се нулира. Акулата започва да се движи из игралното поле – завърта се на случаен принцип и прави 5 стъпки, като при достигане на края на игралното поле се отблъсква и продължава движението си. Стартира се брояч за време от 1 минута.
  - Когато е натиснат клавиш стрелка, малката рибка се обръща и придвижва с 10 стъпки в указаната посока.
  - Когато акулата докосне малката рибка, точките се увеличават с 1, чува се звук. Движението на акулата продължава.
  - При изтичане на времето излиза съобщение „Край“ (фигура 10).



Фигура 10. Бягай от акулата

### Приложение 1. Средата Scratch – кратко въведение

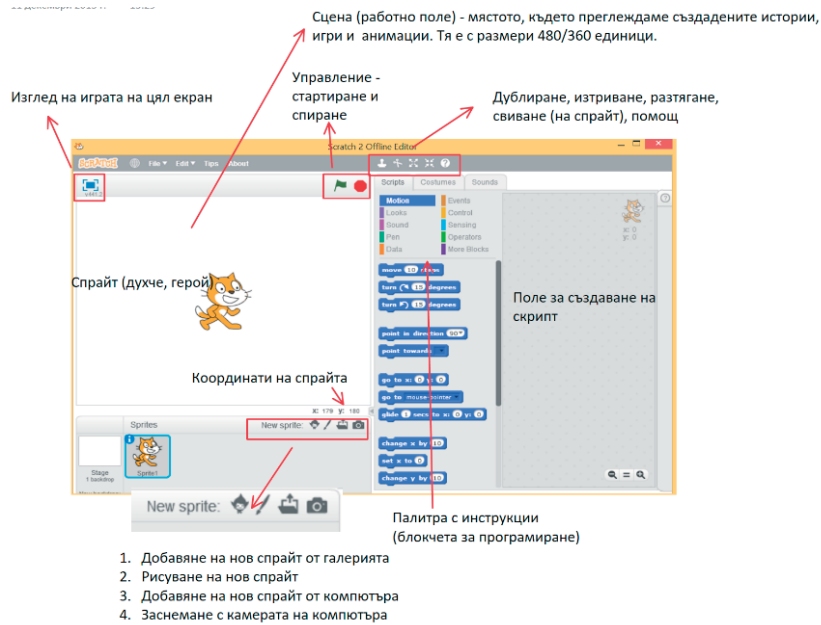
Средата Scratch може да се използва онлайн. Съществува и Scratch 2.0. офлайн редактор, който предлага много удобна работа на потребителя. Редакторът е на разположение за свободно изтегляне и инсталиране за Windows, Mac OS и Linux. Изтеглянето на редактора става от адрес: <http://scratch.mit.edu/download>. Следват се стъпки от 1 до 3 (фиг. 1), като на стъпка 3 се изтеглят допълнителни информационни и помощни материали и примери (помощните материали са на английски език).



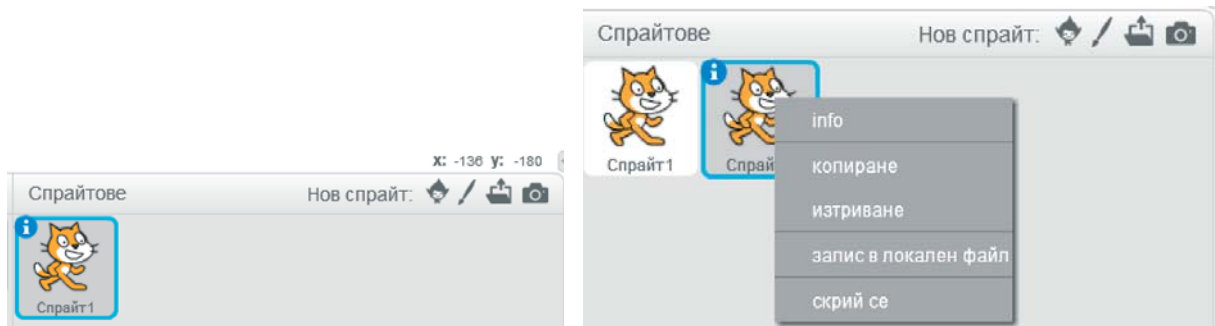
Фигура 1. Инсталиране на Scratch

Потребителският интерфейс на програмата Scratch е описан на фигура 2. Работният екран предлага зона за преглед на създадения проект, зона с командни блокове, зона за създаване блоковата програма и всички нужни инструменти за работа с обекти, фон и т.н. Обектите в средата Scratch се наричат „спрайтове“. Спрайтовете (Sprites) са активни графични обекти в Scratch. Спрайтовете могат да се избират от галерия на програмата, могат да се създават чрез рисуване в редактор на средата, могат да се въведат от готови изображения или да се заснемат, като се използва вградената в компютърната система камера.

За преглед или редактиране на даден спрайт, той трябва да бъде избран чрез щракване върху миниатюрата му или чрез двукратно щракване върху героя на работното поле. Чрез щракване с десен бутон върху миниатюрата на спрайта, той може да бъде корипан, изтрит, записан във файл или скрит. Чрез влачене на миниатюрите можем да пренареждаме спрайтовете (Фигура 3).

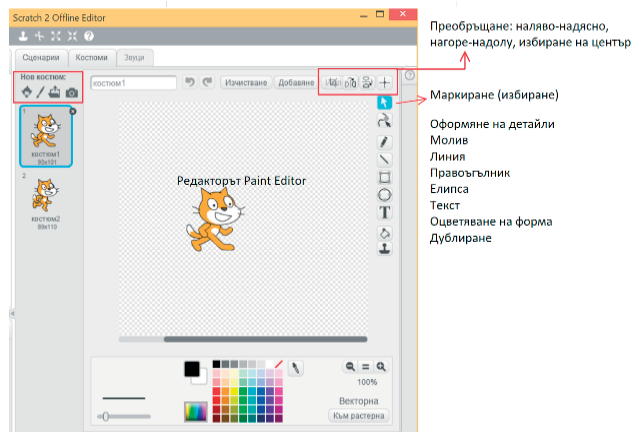


Фигура 2. Интерфейс на средата



Фигура 3. Спрайт

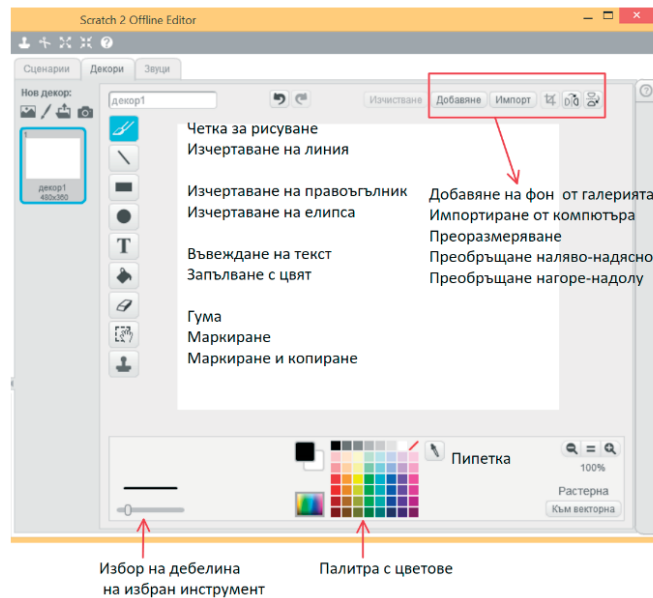
Всеки спрайт може да има различни костюми. Щраквайки върху раздел „Костюми“ виждаме, че нашият спрайт има два костюма. В случая е избран костюм 1. С влачене можем да пренареждаме костюмите. Можем да добавяме и нови костюми. В редакторът Paint Editor, подобно на познатия MS Paint графичен редактор, можем да редактираме костюма на героя или да създадем нов герой (фиг. 4).



Фигура 4. Костюми

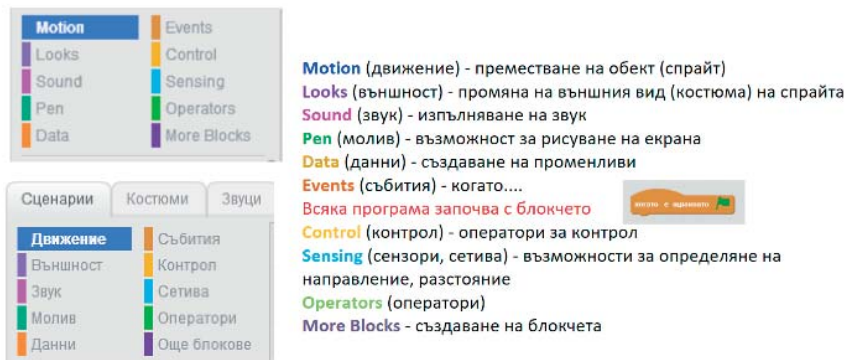
Освен обектите спрайтове, средата поддържа и обект „декор“ (още – сцена, фон). По подразбиране декорът е бял фон. Средата поддържа галерия от готови декори, от които потребителят може да избере. Подобно на спрайтовете, декорът може да се вмъкне от изображение, да се рисува с помощта на вграденния редактор или да се заснеме с помощта на вградената камера (фиг. 5).





Фигура 5. Декори

Създаването на програмата става с помощта на команди, които са визуализирани в средата като градивни блокове. Блоковете са разделени в групи и са с различна цвятова идентификация – команди за движение, команди за определяне на външния вид, команди за управление и т.н. (фиг. 6).



Фигура 6. Командни блокове

Можем да даваме инструкции на героя (спрайта), като му „казваме“ да се движи, да издава звуци или да реагира на друг герой (спрайт). Инструкциите даваме като съединяваме цветните блокчета с команди едно с друго. Така се получават скриптове (още – кодове, програми). За да стартираме скрипта, щракваме двукратно върху него и проследяваме изпълнението на кода в полето за визуализация. Някои блокове имат поле за въвеждане и редактиране на текст, а други – падащи менюта.

#### Notes:

1. <http://www.mon.bg/?go=page&pagelId=7&subpagelId=57> МОН. Закони. ЗАКОН за предучилищното и училищното образование
2. <http://www.mon.bg/?go=page&pagelId=7&subpagelId=59> МОН. Наредби. Наредби 2015. НАРЕДБА № 5 от 30.11.2015 г. за общооб разователната подготовка
3. <http://www.mon.bg/?go=page&pagelId=7&subpagelId=59> МОН. Наредби. НАРЕДБА № 12 от 01.09.2016 г. за статута и професионалното развитие на учителите, директорите и другите педагогически специалисти
4. <http://www.mon.bg/?go=page&pagelId=1&subpagelId=1689> Учебни програми за III клас в сила от учебната 2018-2019 година. Компютърно моделиране.
5. <https://scratch.mit.edu/> Scratch