Учебен сценарий „Почисти морското дъно“

|  |  |
| --- | --- |
| **Сценарии на урок** | Почисти морското дъно (разказване на история) |
| **Предишен опит в програмирането** | Задаване на начални координати  Задаване на размер за спрайт  Добавяне на текст за спрайт  Движение на обект с клавиши със стрелки с помощта на събития  Използването на условен блок с докосване за състоянието на обекта |
| **Резултати от обучението** | **Общи резултати от обучението:**  Основни очаквани резултати:  ● Променливи  ● Показване и скриване на спрайтове  ● Дублирани спрайтове  ● Дублиран блок от код  ● Условни блокове  **Специфични резултати от обучението, ориентирани към алгоритмично мислене:**  Специфични очаквани резултати  • Ученикът използва променлива за преброяване на събраните отпадъци  • Ученикът използва скрит спрайт, когато е докоснат спрайт и показва спрайт в началото  • Ученикът знае как да дублира спрайт (от една бутилка до например 4 бутилки)   * Ученикът знае как да използва условни условия за проверка дали е показан спрайт и дали всички боклуци са взети   • Ученикът прави рефакторинг на съществуващ код |
| **Цели, Задачи и кратко описание на дейностите** | **Кратко описание:**  Проблемът: Морското дъно е пълно с боклук и октоподът решава да го почисти. Когато събере всички боклуци, той ги хвърля в кошчето.  Задачи: Учениците започват със задаване на начални координати за октопода.  Играта приключва, когато октоподът събере всички боклуци и ги сложи в кошчето.  За целта учениците ще трябва да използват променливи за преброяване на точки (1 събран боклук= 1 точка).  Когато октоподът докосне боклук, го вдига, боклукът се скрива и броят точки се увеличава с 1.  Когато вземе всички бутилки и хартия, той отива до кошчето. Ако той не вземе всички боклуци и отиде до кошчето по-рано, кошчето казва да се върне, за да вземе останалия боклук.  **Цел: Учениците ще научат как да използват променливи и как да дублират блок код или дори цял спрайт.** |
| **Продължителност** | 45 минути |
| **Методи на обучение** | Активно обучение, обучение, основано на игрови дизайн, решаване на проблеми |
| **Форма на преподаване** | Самостоятелна работа / Работа по двоики |
| **Ход на урока** | * (Мотивация-Въведение, Прилагане, Осмисляне и Оценка)   Учителят дискутира с учениците проблема с изхвърлянето на боклук в природата. Дискутира се каква е вредата, която произтича от това. Чрез програмиране на игра, учениците ще помогнат на октопод да почисти морското дъно.  Първоначално се дава на учениците:  ● Фон  ● Спрайт за октопода (с кода за движение), спрайт за бутилки, спрайт за хартия и спрайт за кошче.   * Спрайт за Звезда, която ще връща октопода в начална позиция, при докосване на октопода със звездата. Звездата се движи хаотично и октопода трябва да избягва свещата с нея.   Октопода иска да се разходи и да се наслади на деня си Но вижда, че морското дъно е пълно с боклук. Той решава да вземе всички боклуци. Когато го направи, най-накрая може да се наслади на деня си в чиста среда.     * Учителят поставя задача да се стартира и разгледа сцената и героите на проекта колко сцени има, колко герои, колко костюма имат героите.   **Стъпка 1**   * Даден е фонът, а също и спрайта на октопода с код за движение с клавиши и условието за докосване на звездата.       **Стъпка 2**   * Учителят поставя задача как да преброим взетия боклук. За преброяване на броя на събрания боклук, ще използваме променливи. * Какво е променлива? * Променливата е като кутия, където съхраняваме някаква информация. * В нашия случай можем да видим нашата променлива като поле, наречено точки. Когато октоподът вземе боклук, броят му се съхранява в променливи точки. Тази променлива отчита колко боклук е събрал октопода. * Как да направим променлива?     /Users/mateja.bevcic/Desktop/Learning Scenario/igre/3_spremenljivke_smeti/7.png  Избираме оранжев блок Променливи, след това щракваме върху бутона Създаване на променлива, пишем име на променлива и натискаме OK. След това се появяват блок points, като поставяме отметка в квадратчето.  /Users/mateja.bevcic/Desktop/Learning Scenario/igre/3_spremenljivke_smeti/8.png  Ако квадратчето е отметнато, променливата с нейната стойност ще се вижда на екрана:    В началото на играта стойността на променливата трябва да бъде 0, тъй като няма прибран боклук. Под кода от [Стъпка 1] ученикът добавя набор от блокове настрой\_\_ до 0. Чрез кликване върху падащото меню те избират подходяща променлива, която е за точките.    **Стъпка 3**   * Учениците пишат код за бутилка. Идеята е, че спрайтът изчезва (което означава скрий), когато докосне октопода. * Така кодът ще започне, когато боклук докосне октопода. Тогава трябва да помислим в кой случай той взема боклука. Ако спрайтът (бутилката) все още е там, ние го вдигаме „и го поставяме в полето с променливи“. Преди имахме 0 елемента в променливите точки, сега имаме 1. Можем да видим, че като вземем боклука, променяме броя на променливите (точки) с 1, което е, увеличаваме с 1. Когато боклукът бъде взет, ние го скриваме.      * Коментират се необходимите блокове * Сега можем да тестваме дали нашият код е правилен. * Щракаме върху зелено знаме и взимаме бутилката. Бутилката трябва да изчезне и броят точки трябва да бъде 1. * След това искаме да играем отново играта и отново щракваме върху зеления флаг. Какво става? Къде е бутилката сега? * Бутилката е скрита, ние я скрихме преди. Така че в началото на играта трябва да програмираме, че бутилката се показва. Правим това, като изберем блок Покажи.     Стъпка 4  учениците искат да имат повече бутилки в играта си, затова трябва да дублират своя спрайт. Те щракват с десния бутон върху спрайта и избират дубликация.     * Учениците трябва да щракнат с мишката върху новата бутилка и да я плъзнат някъде в сцената. * Те могат да повторят тази стъпка и да дублират бутилката отново.   Учениците искат да имат същия код за хартиения спрайт. Те могат да дублират кода на бутилката, като щракнат с десния бутон върху блока с код:     * Пуснете дублирания код в хартиения спрайт, като щракнете с мишката върху него. * Учениците повтарят тази стъпка, за да дублират блока с код, когато е щракнат зеления флаг - покажи. * Те могат също да повторят [Стъпка 4] и да дублират целия хартиен спрайт, за да има повече боклук от хартия в сцената.   **Стъпка 5**   * Последното нещо, което учениците трябва да направят, е да напишат код за кошчето. Кошчето е дадено, те могат да го преместят навсякъде вътре в морското дъно. * Също така този код ще се активира, когато октоподът го докосне. * Кошчето ще трябва да провери дали всички боклуци са взети. Благодарение на променливата за точки това ще бъде лесно да се направи. Да приемем, че имаме 8 спрайта в играта, така че учениците трябва да проверят дали броят на точките е равен на 8. Ако е, това означава, че всички боклуци са взети, в противен случай не е така. Те ще използват if, за да програмират това и ще добавят малко текст, за да кажат на играча дали е взел всички боклуци или не.     Учителят поставя задачата учениците да направят монолога на Октопода при стартрирането на историята.  Дискусия как могат да синхронизират действията с задаване време на всеки от героите.  На учениците се дава да разгледат кода на част от проекта  Учениците трябва да зададат началните координати за октопода с блок x, y. Координатите се избират сами, важно е само да са на пътя. Учениците вече знаят как да задават координатите от предишни дейности. Те също така добавят някои инструкции.    Обсъжда се групово как да се довърши историята от картинката.  Финален код  Октопод (Girl)      Бутилки (Bottle) / Хартия (Paper)    Кошче (Trash)    Звезда (Star)    Допълнителни задачи  Учениците могат да добавят допълнителни задачи според техните желания или могат да следват задачите по-долу:  ● Добавете друг вид отпадъци (например биоотпадъци).  ● В кошчето пише напр. „Взехте X бутилки, Y хартии и Z дини“.  ● Ако играч вземе цялото кошче, кошчето казва: „Поздравления! Ти почисти морското дъно! “  ● Ако играчът не вземе цялото кошче, кошчето му казва кой боклук не е бил взет, напр. „Не сте взели всички бутилки. Не сте взели всички дини. и „Върнете се, когато вземете всички боклуци“.   * Учителят демонстрира проектът. * Учителят коментира, че за да се разкаже една история най-напред трябва да се направи нейния сценарии. За целта може да се използва допълнителна таблица за описание на сценария на историята. По преценка на учителя може да се даде готовата таблица или да се даде частично попълнена, а учениците базирайки се на демонстрацията да я попълнят. |
| **Ресурси за учителя** | Цялата дейност е в Snap!:  <https://snap.berkeley.edu/snap/snap.html#present:Username=elenagmg&ProjectName=Sea-floor-end> |
| **Ресурси за учениците** | <https://snap.berkeley.edu/project?user=elenagmg&project=Sea-floor-end> |

Приложение 1. Сценарии на историята

Сцени

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Design | Actions | Notes |
| 1. Start |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sprites

|  |  |  |
| --- | --- | --- |
| Sprite | Actions | Stage background |