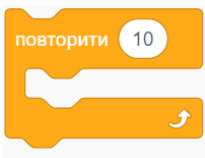
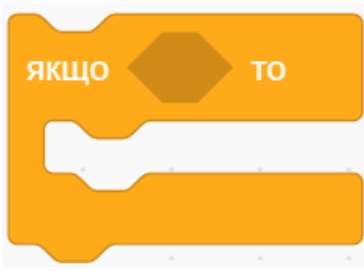
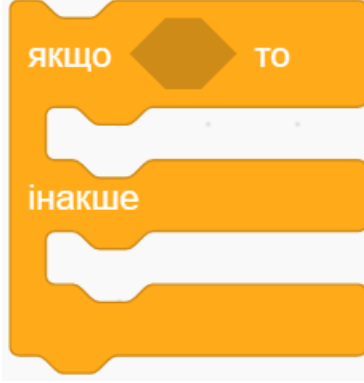
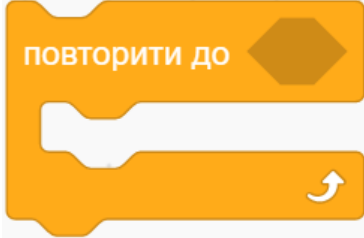
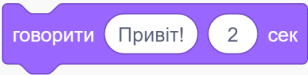
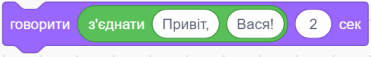
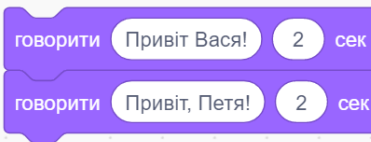
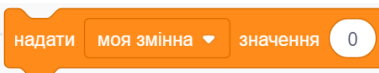
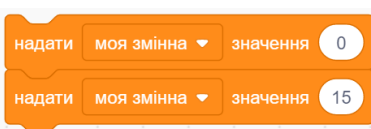
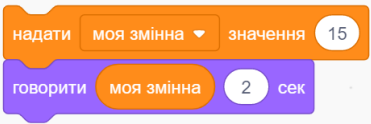
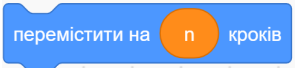
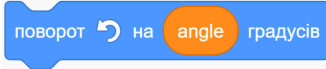
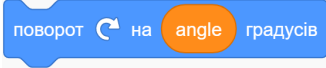
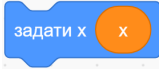
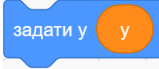
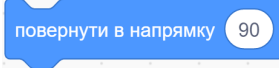
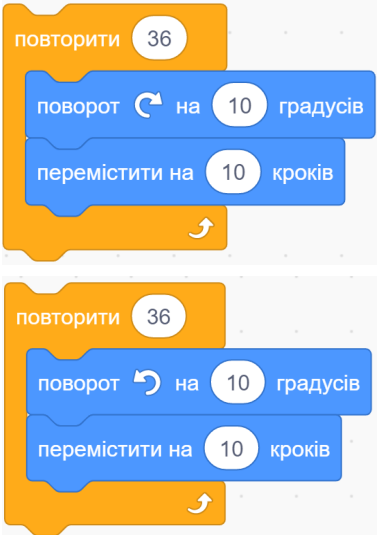
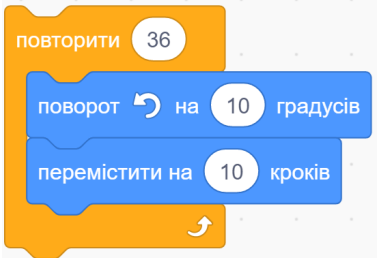
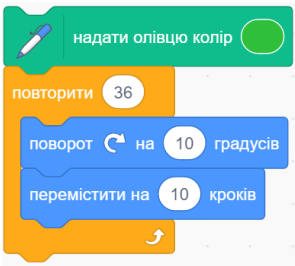
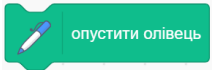
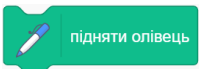
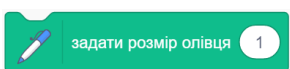
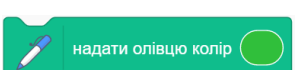
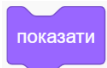
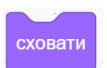
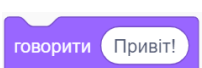


	Пояснення команди	
	Початок виконання алгоритму	-
	Кінець виконання алгоритму	-
		<pre>for x in range(n+x): </pre>
	Неповна форма команди розгалуження	<pre>if умова перша: дія перша</pre>
	Повна форма команди розгалуження	<pre>if умова перша: дія перша else: дія друга</pre>
	Цикл з умовою	<pre>while умова: дія перша дія друга ... дія n</pre>

	Вивід на екран інформацію, В лапках записується текст для виводу	<code>print('Привіт!')</code>
	Вивід кількох символівних рядків на екран інформацію	<code>print('Привіт,', 'Вася!')</code>
	Вивід кількох повідомлень на екран інформацію	<code>print('Привіт Вася!')</code> <code>print('Привіт, Петя!')</code>
	Обчислення суми та виводу результату на екран	<code>print(10+13)</code>
	Обчислення різниці та виводу результату на екран	<code>print(10-13)</code>
	Обчислення добутку та виводу результату на екран	<code>print(10*13)</code>
	Обчислення частки та виводу результату на екран	<code>print(10/13)</code>
	Надання значення змінній	<code>my_variable=0</code>
	Надання змінній іншого значення	<code>my_variable=0</code> <code>my_variable=15</code>
	Надання змінній значення та виводу результату на екран	<code>my_variable=15</code> <code>print(my_variable)</code>
	Вивести на екран інформацію та змінну	<code>print('Текст',my_variable)</code>



 Олівець Малювати за допомогою спрайтів.	Додавання бібліотеки для малювання	<pre>import turtle</pre>
	Проповзти вперед на n пікселів (кроків)	<pre>turtle.forward (n)</pre>
	Проповзти назад на n пікселів (кроків)	<pre>turtle.backward (n)</pre>
	Повернутися наліво на angle градусів	<pre>turtle.right (angle)</pre>
	Повернутися направо на angle градусів	<pre>turtle.left (angle)</pre>
	Перемістити черепашку в точку з координатами (x, y)	<pre>turtle.goto (x, y)</pre>
	Встановити x координату черепашки	<pre>turtle.setx (x)</pre>
	Встановити y координату черепашки	<pre>turtle.sety (y)</pre>
	Повернути черепашку під кутом to_angle до вертикалі (0 - наверх, 90 - направо)	<pre>turtle.setheading (to_angle)</pre>
	Повернути черепашку додому - в точку, з координатами (0,0)	<pre>turtle.home ()</pre>

	Намалювати коло радіуса r, центр якої знаходиться зліва від черепашки, якщо $r > 0$ і справа, якщо $r < 0$	<pre>turtle.circle (radius)</pre>
		
	Намалювати точку діаметра size кольору color. Параметр color необов'язковий	<pre>turtle.dot (size, color)</pre>
	Почати малювати	<pre>turtle.pendown ()</pre>
	Закінчити малювати	<pre>turtle.penup ()</pre>
	Встановити діаметр пера	<pre>turtle.pensize (width)</pre>
	Встановити колір лінії, яка малює черепашка	<pre>turtle.pencolor (colorstring)</pre>
-	Встановити колір заповнення	<pre>turtle.fillcolor (colorstring)</pre>
	Показати черепашку	<pre>turtle.showturtle ()</pre>
	Сховати черепашку	<pre>turtle.hideturtle ()</pre>
	Вивести текст text	<pre>turtle.write (text)</pre>

-	Отримати поточні координати черепашки	<code>turtle.position ()</code>
значення x	Отримати x координату черепашки	<code>turtle.xcor ()</code>
значення y	Отримати y координату черепашки	<code>turtle.ycor ()</code>
 ОЧИСТИТИ ВСЕ	Очистити поле	<code>turtle.clear()</code>