

# Turtle Plots

# Moving the Turtle

---

- Idea: trace walk-path

# Moving the Turtle

- Idea: trace walk-path

## C++ Commands

- Step (drawn): `turtle::forward();`
- Step (not drawn): `turtle::jump();`
- Rotation left: `turtle::left(my_angle);`
- Rotation right: `turtle::right(my_angle);`
- Save position: `turtle::save();`
- Load position: `turtle::restore();`
- Change color: `turtle::colorcycle();`

Requires: `#include "turtle.h"`

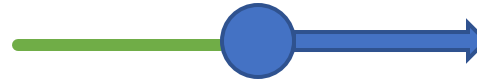
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



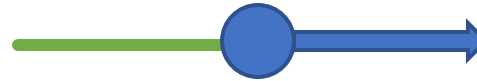
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



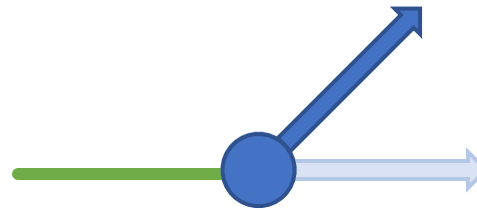
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



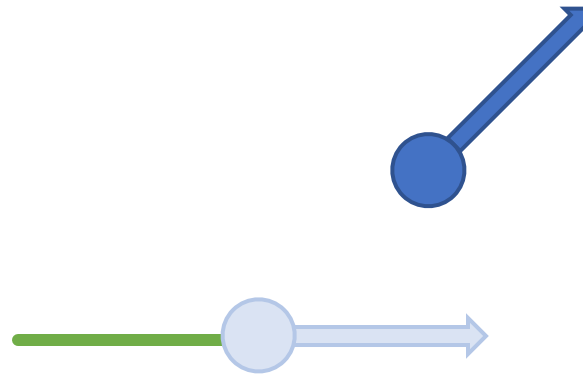
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



# Moving the Turtle

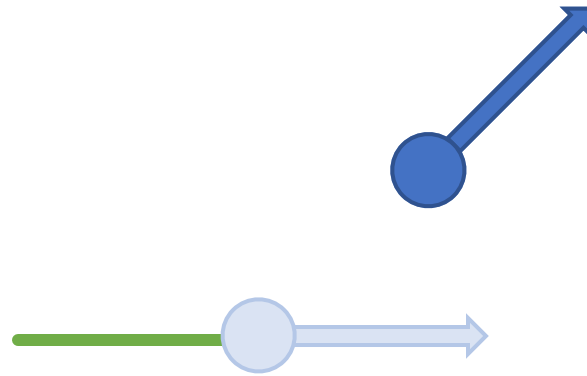
```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```





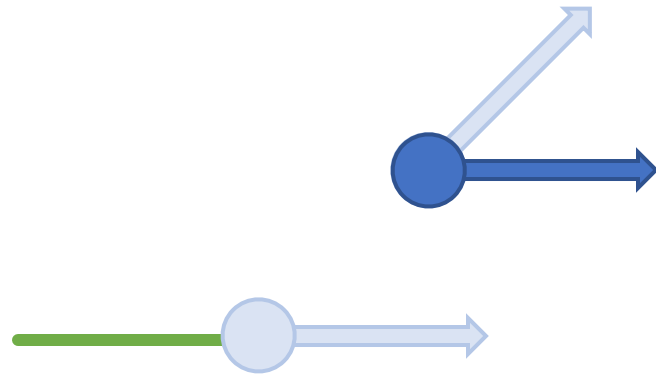
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



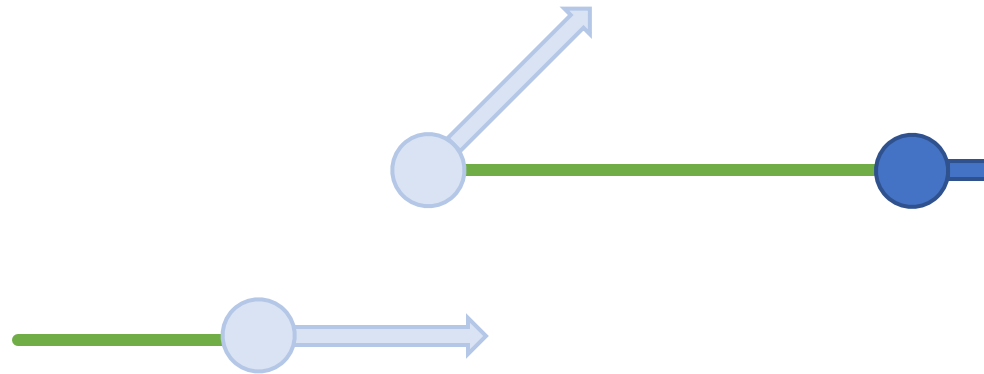
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



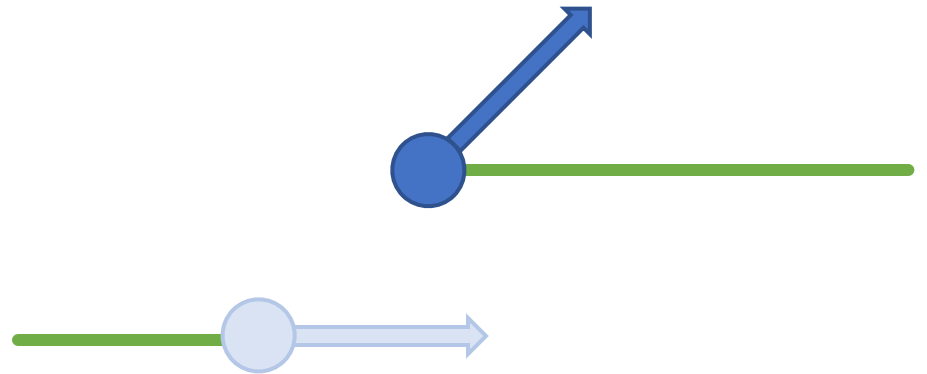
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



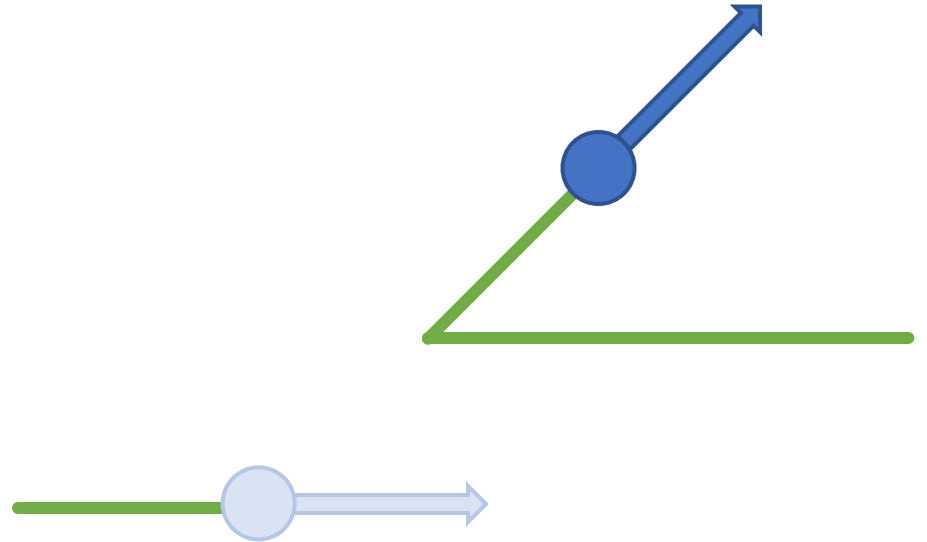
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



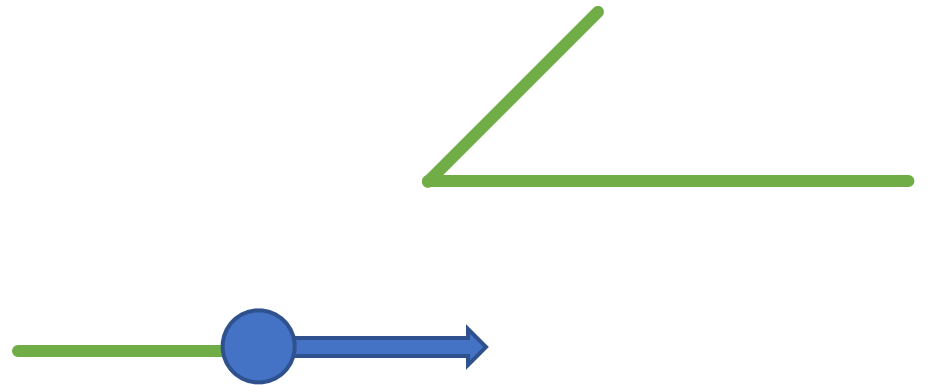
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



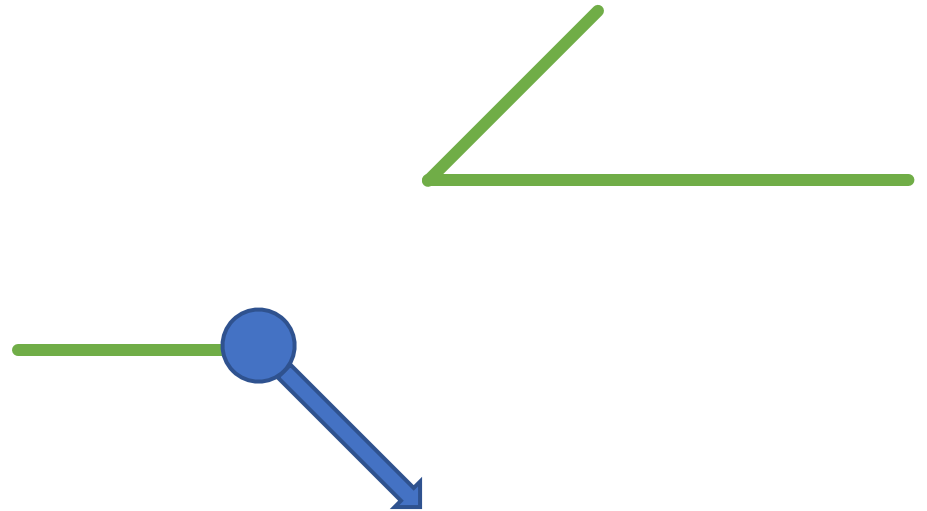
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



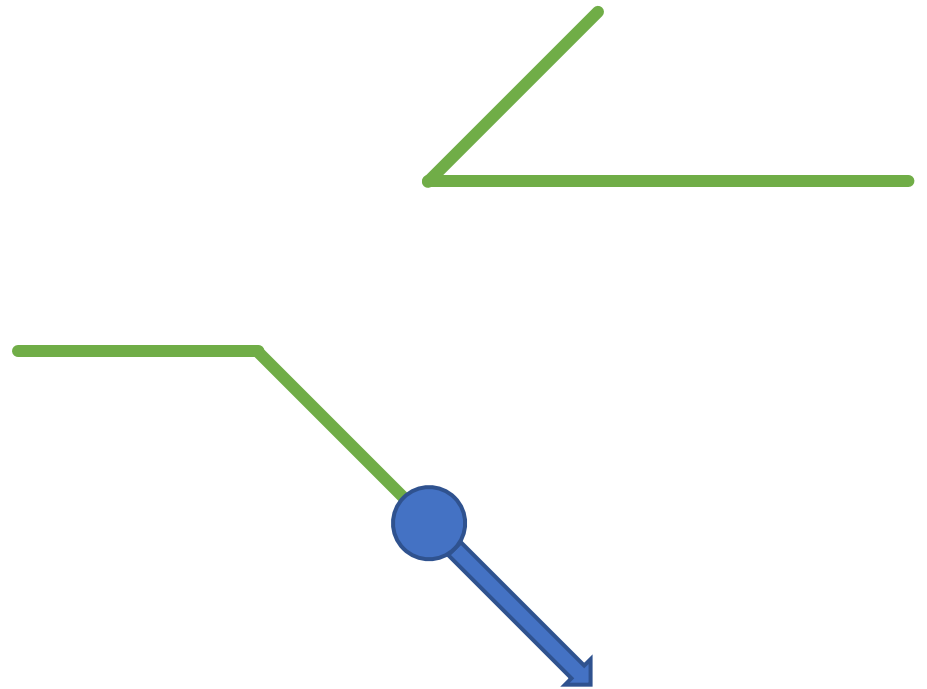
# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



# Moving the Turtle

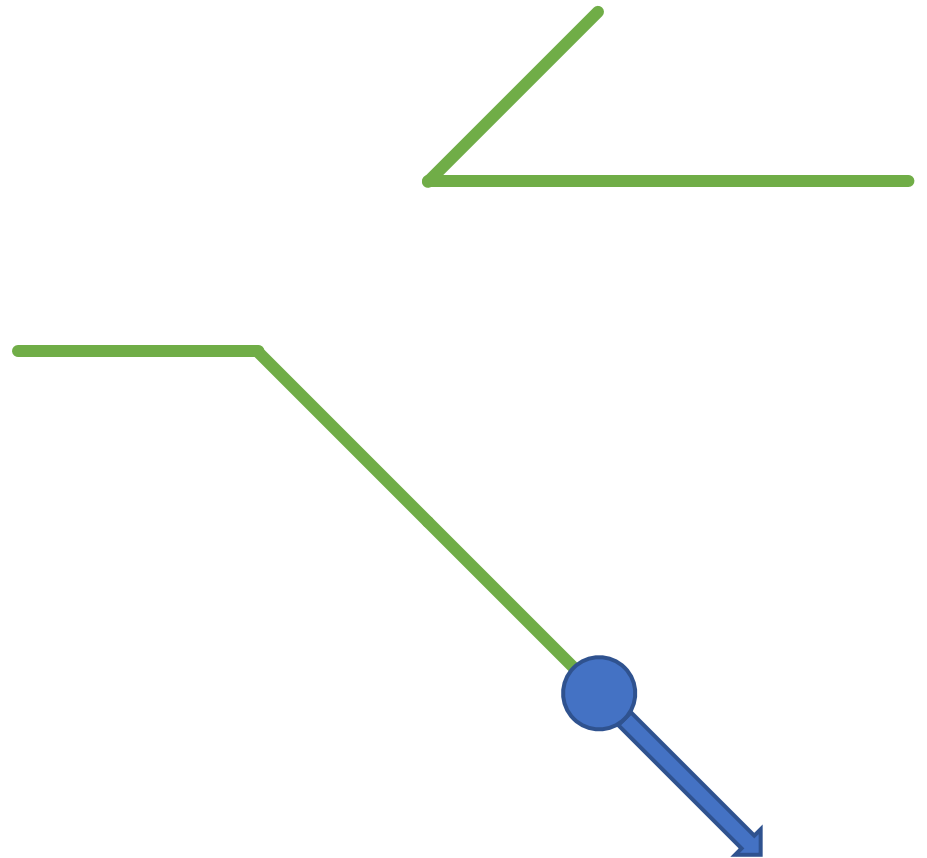
```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```





# Moving the Turtle

```
turtle::forward();  
turtle::save();  
turtle::left(45);  
turtle::jump();  
turtle::save();  
turtle::right(45);  
turtle::forward(2);  
turtle::restore();  
turtle::forward();  
turtle::restore();  
turtle::right(45);  
turtle::forward();  
turtle::forward();
```



colorcycle

# colorcycle

---

- `turtle::colorcycle()` ;  
    → changes drawing color of turtle
- Cycles through colors
  - call it 200 times to return to initial color

# colorcycle

---

- `turtle::colorcycle()` ;  
    → changes drawing color of turtle
- Cycles through colors
  - call it 200 times to return to initial color
- `turtle::colorcycle2(val)` ;
  - increment color by `val` (type double)
  - `turtle::colorcycle()` uses 0.005 for `val`

# colorcycle-Example

## Example:

```
#include "turtle.h"

int main ()
{
    turtle::left(45);
    for (int i = 0; i < 100; ++i) {
        turtle::forward();
        turtle::colorcycle();
        turtle::colorcycle();
    }

    return 0;
}
```

