

```

    }
    Code to execute while condition is true
}
if (analog (port#)>=#)
//Check sensor value in port #, if this condition is met, execute condition in curly braces
{
    Code to execute while condition is true
}
while (digital (port#)==0) or
//Check sensor value in port #, if this condition is true execute condition
while (a_button ()==0) or

> Greater than      < Less than      == Equal to      != Not equal to      && and      || or
<= Less than or equal

```

```

printf("text\n");
motor(port#, % power);
msleep (# milliseconds);
ao();
enable_servos();
set_servo_position(port#, position);
disable_servos ();
digital(port #);
analog (port#);
analog_et(port#);
get_motor_position_counter(port#);
clear_motor_position_counter(port#);

//Prints text to display
//Turns motor on at % power specified
//Program waits specified number of milliseconds
//All off, turns all motor ports off
//Turns servo ports on
//Moves servo in specified port to a set position
//Turns off servo ports
//Refers to a specific digital port #
//Refers to a specific analog port #
//Get an analog ET sensor reading
//Get the position of the motor in the specified port
//Clear the position of the motor in the specified port

```

```

    }
    Code to execute while condition is true
}
if (analog (port#)>=#)
//Check sensor value in port #, if this condition is met, execute condition in curly braces
{
    Code to execute while condition is true
}
while (digital (port#)==0)
//Check sensor value in port #, if this condition is true execute condition
while (a_button ()==0) or

> Greater than      < Less than      == Equal to      != Not equal to      && and      || or
<= Less than or equal

```

```

printf("text\n");
motor(port#, % power);
msleep (# milliseconds);
ao();
enable_servos();
set_servo_position(port#, position);
disable_servos ();
digital(port #);
analog (port#);
analog_et(port#);
get_motor_position_counter(port#);
clear_motor_position_counter(port#);

//Prints text to display
//Turns motor on at % power specified
//Program waits specified number of milliseconds
//All off, turns all motor ports off
//Turns servo ports on
//Moves servo in specified port to a set position
//Turns off servo ports
//Refers to a specific digital port #
//Refers to a specific analog port #
//Get an analog ET sensor reading
//Get the position of the motor in the specified port
//Clear the position of the motor in the specified port

```