

Arduino C++ Programming

Advanced Concepts

pow()

```
z = pow(x, y);
```

Returns double
X and y are floats

Math Functions (read docs for full specs)

abs()

abs()

```
x = -42;  
Serial.print("The absolute value of ");  
Serial.print(x);  
Serial.print(" is ");  
Serial.println(abs(x));
```

sq()

```
int input = Serial.parseInt();  
int inputSquared = sq(input);
```

sqrt()

```
sqrt(x)
```

map()

```
val = map(val, 0, 1023, 0, 255);
```

max() and min()

```
sensVal = max(sensVal, 20); // assigns sensVal to the larger of sensVal or 20  
                        // (effectively ensuring that it is at least 20)  
sensVal = min(sensVal, 100); // assigns sensVal to the smaller of sensVal or 100  
                        // ensuring that it never gets above 100.
```

map()

The Arduino `map()` function takes a number from one numeric range and proportionally converts it to an equivalent value within another specified numeric range.

```
map(value, fromLow, fromHigh, toLow, toHigh);
```

Parameters:

`value`: The number to be mapped.

`fromLow`: The lower bound of the original range.

`fromHigh`: The upper bound of the original range.

`toLow`: The lower bound of the target range.

`toHigh`: The upper bound of the target range.

Random

```
long randNumber;

void setup() {
  Serial.begin(9600);

  // if analog input pin 0 is unconnected, random analog
  // noise will cause the call to randomSeed() to generate
  // different seed numbers each time the sketch runs.
  // randomSeed() will then shuffle the random function.
  randomSeed(analogRead(0));
}

void loop() {
  // print a random number from 0 to 299
  randNumber = random(300);
  Serial.println(randNumber);

  // print a random number from 10 to 19
  randNumber = random(10, 20);
  Serial.println(randNumber);

  delay(50);
}
```

Time

delay()

```
digitalWrite(ledPin, HIGH); // sets the LED on
delay(1000);                // waits for a second
digitalWrite(ledPin, LOW);  // sets the LED off
```

millis()

```
unsigned long myTime;

void setup() {
  Serial.begin(9600);
}

void loop() {
  Serial.print("Time: ");
  myTime = millis();

  Serial.println(myTime); // prints time since program started
  delay(1000);            // wait a second so as not to send massive amounts of data
}
```

Questions and Exercises

1. Create a program to print the square of all numbers from 1 to 10.
2. Modify the last program so that there is a half second delay between print outs of each number and their square.
3. Create a program to display 10 random numbers between 1 and 100.
4. Create a program that creates a random number between 1 and 100 and maps its it to a range of numbers between 1000 and 2000. Print out the original and mapped number next to each other.
5. Ask someone for the diameter of a sphere. Print out its volume. Find the formula for the volume of a sphere on the internet.
6. Create a program that asks a user for 3 numbers and then once retrieved prints out their average and the total time it took for the program to complete in seconds.
7. Create a program to print the first 1000 prime numbers and then displays how long it took to do this in milliseconds.

1. Create a program to print the square of all numbers from 1 to 10.

```
Serial.begin(9600);
```

```
for (int i=1; i<11; i++){
```

```
    int sqr = sq(i);
```

```
    Serial.println(sqr);
```

```
}
```

2. Modify the last program so that there is a half second delay between print outs of each number and their square.

```
Serial.begin(9600);
```

```
for (int i=1; i<11; i++){  
    int sqr = sq(i);  
    Serial.println(sqr);  
    delay(500);  
}
```


3. Create a program to display 10 random numbers between 1 and 100.

```
Serial.begin(9600);  
  randomSeed(analogRead(1));  
  
for (int i=1; i<11; i++){  
  int randomNum = random(101);  
  Serial.println(randomNum);  
}
```

4. Create a program that creates a random number between 1 and 100 and maps it to a range of numbers between 1000 and 2000. Print out the original and mapped number next to each other.

```
Serial.begin(9600);
randomSeed(analogRead(1));

for (int i=1; i<11; i++){
  int randomNum = random(101);
  Serial.println(randomNum);

  int mappedNum = map(randomNum, 1, 100, 1000, 2000); // Map to range 1000-2000

  // Print both numbers
  Serial.print("Original number: ");
  Serial.print(randomNum);
  Serial.print(" | Mapped number: ");
  Serial.println(mappedNum);
}
```

5. Ask someone for the diameter of a sphere. Print out its volume. Find the formula for the volume of a sphere on the internet.

```
Serial.begin(9600);
```

```
Serial.println("Enter diameter: ");
```

```
while(Serial.available()==0){}
```

```
float dia = Serial.parseFloat();
```

```
Serial.println(dia);
```

```
double diaCubed = pow(dia, 3);
```

```
Serial.println(diaCubed);
```

```
// Note: the 1/6 must be float because otherwise
```

```
// Int division makes it 0 not 0.blah blah
```

```
double volume = (1.0/6.0)*(3.14)*(diaCubed);
```

```
Serial.println(volume);
```

6. Create a program that asks a user for 3 numbers and then once retrieved prints out their average and the total time it took for the program to complete in seconds.

```
Serial.begin(9600);  
Serial.println("Enter num1: ");  
while(Serial.available()==0){}  
int num1 = Serial.parseInt();  
  
Serial.println("Enter num2: ");  
while(Serial.available()==0){}  
int num2 = Serial.parseInt();  
  
Serial.println("Enter num3: ");  
while(Serial.available()==0){}  
int num3 = Serial.parseInt();  
  
double average = (num1+num2+num3)/3.0;  
  
Serial.println("Average: "+String(average));  
  
myTime = (millis())/1000;  
Serial.println("Time taken: "+String(myTime));
```

7. Create a program to print the first 1000 prime numbers and then displays how long it took to do this in milliseconds.

```
Serial.begin(9600);
long startTime = millis();

for (int i = 2; i <= 1000; i++) {
    bool isPrime = true;

    for (int j = 2; j * j <= i; j++) {
        if (i % j == 0) {
            isPrime = false;
            break; // correctly break inner loop if number is not prime
        }
    }

    if (isPrime) {
        Serial.println(i); // print prime number
    }
}

float elapsedTime = (millis() - startTime)/1000.0;
Serial.println("Time taken: "+String(elapsedTime));
```