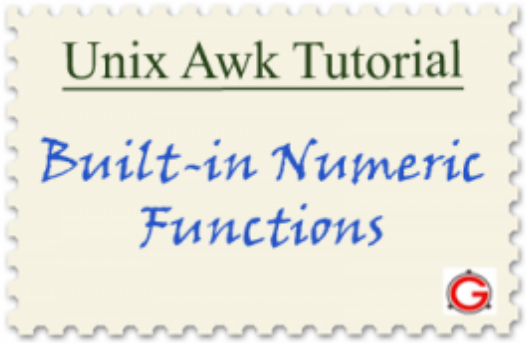
9 Powerful Awk Built-in Functions for Numeric

by SASIKALA *on* MARCH 17, 2010



Similar to awk built-in variables, awk also has lot of built-in functions for numeric, string, input, and ouput operations. Awk has the following three types of high level built-in function categories.

1. Built-in functions for numeric operations
2. Built-in functions for String operations
3. Built-in functions for Input Output operations

For those who are new to awk, please refer to our on-going awk examples series, where we discussed about — awk introduction, awk variables, and awk operators.

In this article, let us review awk Numeric built-in functions.

1. Awk int(n) Function

Example

```
$ awk 'BEGIN{
print int(3.534);
print int(4);
print int(-5.223);
print int(-5);
}'
3
4
-5
-5
```

2. Awk log(n) Function

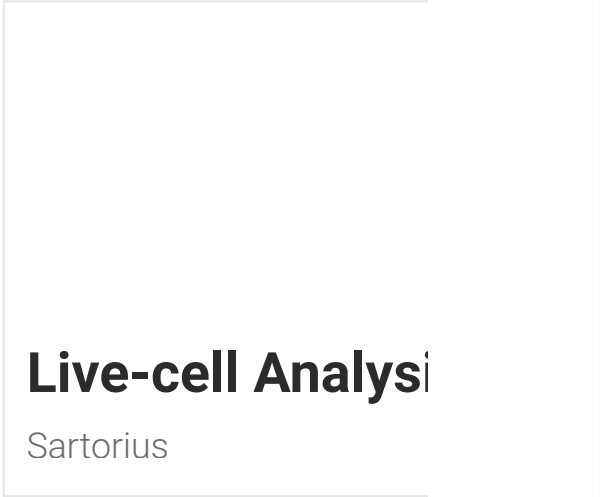
log() function provides natural logarithmic of given argument n. log() returns logarithm value only when n is positive number. If you give any invalid number (even negative) it throws an error.

Example

```
$ awk 'BEGIN{
print log(12);
print log(0);
print log(1);
print log(-1);
}'
2.48491
-inf
0
nan
```

In the above output you can identify that log(0) is infinity which was shown as -inf, and log(-1) gives you the error nan (Not a Number)

argument.



Example

```
$ awk 'BEGIN{
print sqrt(16);
print sqrt(0);
print sqrt(-12);
}'
4
0
nan
```

4. Awk exp(n) Function

exp function provides e to the power of n.

Example

```
$ awk 'BEGIN{
print exp(123434346);
print exp(0);
print exp(-12);
}'
```

In the above output, for exp(1234346), it gives you the output infinity, because this is out of range.

5. Awk sin(n) Function

sin() function gives sine value of n, with n in radians.

Example

```
$ awk 'BEGIN {
print sin(90);
print sin(45);
}'
0.893997
0.850904
```

6. Awk cos(n) Function

cos() returns cosine value of n, with n in radians.

Example

```
$ awk 'BEGIN {
print cos(90);
print cos(45);
}'
-0.448074
0.525322
```

7. Awk atan2(m,n) Function

This function gives you the arc-tangent of m/n in radians.

Example

```
}'
0.588003
```

8. Awk rand() Function

rand() is used to generate the random number between 0 and 1. It never return 0 and 1. It always returns the value between 0 and 1. Numbers are random with in one awk run, but predictable from run to run. Awk uses some algorithm to generate the random numbers. Since this algorithm is fixed, the numbers are repeatable.

Example

The following example generates 1000 random numbers between 0 to 100 and shows how often each number was used

```
$cat rand.awk
BEGIN {
while(i<1000)
{
    n = int(rand()*100);
    rnd[n]++;
    i++;
}
for(i=0;i<=100;i++) {
    print i,"Occured", rnd[i], "times";
}
}
$
```

Pasted some of the output of the above script here.

```
$awk -f rand.awk
0 Occured 6 times
1 Occured 16 times
2 Occured 13 times
```

```
6 Occured 8 times
7 Occured 7 times
8 Occured 16 times
9 Occured 9 times
10 Occured 6 times
11 Occured 9 times
12 Occured 17 times
13 Occured 12 times
```

From the above output, sure that rand() function can generate repeatable numbers very often.

9. Awk srand(n) Function

srand() is used to initialize the random generation with the given argument n. So that whenever the program execution starts, it starts generating the number from n. If no argument is given, it uses the time of the day to generate the seed.

Example. Generate 5 random number starting from 5 to 50

```
$cat srand.awk
BEGIN {
#initialize the seed with 5.
srand(5);
# Totally I want to generate 5 numbers.
total=5;
#maximum number is 50.
max=50;
count=0;
while(count < total) {
    rnd = int(rand() * max);
    if ( array[rnd] == 0 ) {
        count++;
        array[rnd]++;
    }
}
```

```
}
}
```

In this srand.awk, using rand() function, generate the number and multiply with max value to produce the number with the max of 50, and check if the generated random number is already exist in the array, if it does not exist, increment its index and as well as increment loop count. so that it generates 5 number like this and finally in the for loop from minimum number to maximum, and prints the index only which has the value.

Here is the output of the above script

```
$ awk -f  srand.awk
9
15
26
37
39
```


If you enjoyed this article, you might also like..

1. 50 Linux Sysadmin Tutorials
2. 50 Most Frequently Used Linux Commands (With Examples)
3. Top 25 Best Linux Performance Monitoring and Debugging Tools
4. Mommy, I found it! – 15 Practical Linux Find Command Examples
5. Linux 101 Hacks 2nd Edition eBook Free

- Awk Introduction – 7 Awk Print Examples
- Advanced Sed Substitution Examples
- 8 Essential Vim Editor Navigation Fundamentals
- 25 Most Frequently Used Linux IPTables Rules Examples
- Turbocharge PuTTY with 12 Powerful Add-Ons