# BHao_Assign7

## Problem Set 1

Initially, my function did not agree with the sd function because the sd function uses n-1 in the denominator; I updated my function to match.

```
mean_sd = function(x) {
  mean_value = sum(x) / length(x)
  sd_value = (sum((x - mean_value)^2) / (length(x) - 1))^0.5
  return(list(mean = mean_value, sd = sd_value))
}

x = c(1,2,3,4,5)
mean_sd(x)
```

```
## $mean
## [1] 3
##
## $sd
## [1] 1.581139
```

```
mean(x)
```

```
## [1] 3
```

```
sd(x)
```

```
## [1] 1.581139
```

```
mean_sd_running = function(x) {
  # define global variable for expanding array x
  if (!exists('x_running')) {
    assign('x_running', x, envir = .GlobalEnv)
  } else {
    assign('x_running', c(x_running, x), envir = .GlobalEnv)
  }

  return(list(mean = mean(x_running), sd = sd(x_running)))
}

mean_sd_running(x)
```

```
## $mean
## [1] 3
##
## $sd
## [1] 1.581139
```

```
y = c(6,7,8,9,10)
mean_sd_running(y)
```

```
## $mean
## [1] 5.5
##
## $sd
```

```
## [1] 3.02765
```

```
z = c(11,12,13,14,15)
mean_sd_running(z)
```

```
## $mean
## [1] 8
##
## $sd
## [1] 4.472136
```

```
# this time without cheating by using global variables
mean_running = function(x1, n0, mu0) {
  mu = (sum(x1) + n0 * mu0) / (length(x1) + n0)
  return(mu)
}

sd_running = function(x1, n0, mu0, sig0) {
  mu = mean_running(x1, n0, mu0)
  ssdiff0 = sig0^2 * (n0 - 1) + n0 * (mu - mu0)^2  # n-1 in denominator
  ssdiff = ssdiff0 + sum((x1 - mu)^2)
  sd = sqrt(ssdiff / (length(x1) + n0 - 1))  # n-1 in denominator
  return(sd)
}

mean_running(y, length(x), mean(x))
```

```
## [1] 5.5
```

```
sd_running(y, length(x), mean(x), sd(x))
```

```
## [1] 3.02765
```

```
xy = c(x, y)
mean_running(z, length(xy), mean(xy))
```

```
## [1] 8
```

```
sd_running(z, length(xy), mean(xy), sd(xy))
```

```
## [1] 4.472136
```