Homework 1

Vincent

2023-01-16

Question 1

a) "**Sometimes you have to make the hardest climb to see the most beautiful sunrise.** I read that once on an old lady's decorative pillow, but it is really how I feel today. I've climbed a very weird and rocky mountain, and it was a pain in the ass, and my legs are tired, and I'm starving. But *the sun is rising over a sea of love and waffles and possibility.* So *I'm just gonna relax and take a deep breath and enjoy this view for as long as I possibly can."* – Leslie Knope (from ***Parks and Recreation***)

b)

c)

- 2022
  - Fall
    - Stats 15
    - Stats 100A
    - Hist 8A
  - Winter
    - Stats 20
    - Math 115A
    - CS 32
    - CS M51A

d) 

Question 3

```
(6^8)/(16^5)
```

```
## [1] 1.601807
```

```
101^(-8/7)
```

```
## [1] 0.005120908
```

```
8+(3*pi)-3+((4/2)*(-1))
```

```
## [1] 12.42478
```

```
log(54,base =2)
```

```
## [1] 5.754888
```

```
exp(exp(sqrt(2)))
```

```
## [1] 61.14514
```

```
(exp(pi)+exp(-pi))/2
```

```
## [1] 11.59195
```

```
119%/%25
```

```
## [1] 4
```

```
119%%25
```

```
## [1] 19
```

Question 4

```
vol_1 <- (4/3)*pi*(1^3)
vol_4 <- (4/3)*pi*(4^3)

vol_sphere <- function(r=1){
(4/3)*pi*(r^3)
}


vol_sphere()==vol_1
```

```
## [1] TRUE
```

```
vol_sphere(4)==vol_4
```

```
## [1] TRUE
```

Question 5

```
z_prop<- function(x,n,p0){
  ((x/n)-p0)/(sqrt((p0*(1-p0))/n))
}

z_prop(10,13,0.6)
```

```
## [1] 1.245505
```

The z-value 1.245505 means that the sample mean is 1.245505 standard deviation from the population mean.

```
z_prop(30,39,0.6)
```

```
## [1] 2.157277
```

The z-value has increased from 1.245505 to 2.157277. The sample mean increases a lot compare to the sample mean in part b.

Question 6

```
lease_calc <- function(msrp,price,down,n=36,res=0.6,mf=0.001,tax=0.095){
  cap_cost<-price-down
  res_val <- msrp*res
  monthly_dep <- (cap_cost-res_val)/n
  monthly_fine_char <- (cap_cost+res_val)*mf
  sub_total <- monthly_dep+monthly_fine_char
  total <- sub_total+(sub_total*tax)
  total
}

lease_calc(31495,29895,2500,36,0.52,0.0016,0.095)
```

```
## [1] 411.8079
```

Question 7

a) the minimum number of coins required to equal 47 cents is 0.25+0.1*2+0.01*2, 5 coins in total. To get this number, I first divide the total by the largest coin and keep the reminder, then use the reminder to divide the following large coin, and so on.

b) To get the number above, I first divide the total by the largest coin and keep the reminder, then use the reminder to divide the following large coin, and so on.

c)

```
num_of_coins<- function(n){
  coin <- c(0.25,0.1,0.05,0.01)
  count<-1
  for(x in coin[1:4]){
    count <- count+(n%/%x)
    n <- n%%x
  }
  count
}


num_of_coins(0.47)
```

```
## [1] 5
```

d)

```
num_of_coins(0.31)
```

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

```
num_of_coins(0.48)
```

```
## [1] 6
```

```
num_of_coins(1.39)
```

```
## [1] 10
```

e

My logic of finding the number of cents that requires the most coins is to maximize all type of coin. Where the amount of money can't by substitute by a larger coin. $4 * 0.01 + 0.05 + 0.1 * 4 + 0.25 * 2 = 0.99$. And this requires 9 coins

```
for(x in c(1:99)){
  y<-0
  if(num_of_coins(x*0.01)>y){
    y<-x
  }
  else{
    y<-y
  }
}
print(y*0.01)
```

```
## [1] 0.99
```

```
num_of_coins(0.99)
```

```
## [1] 9
```