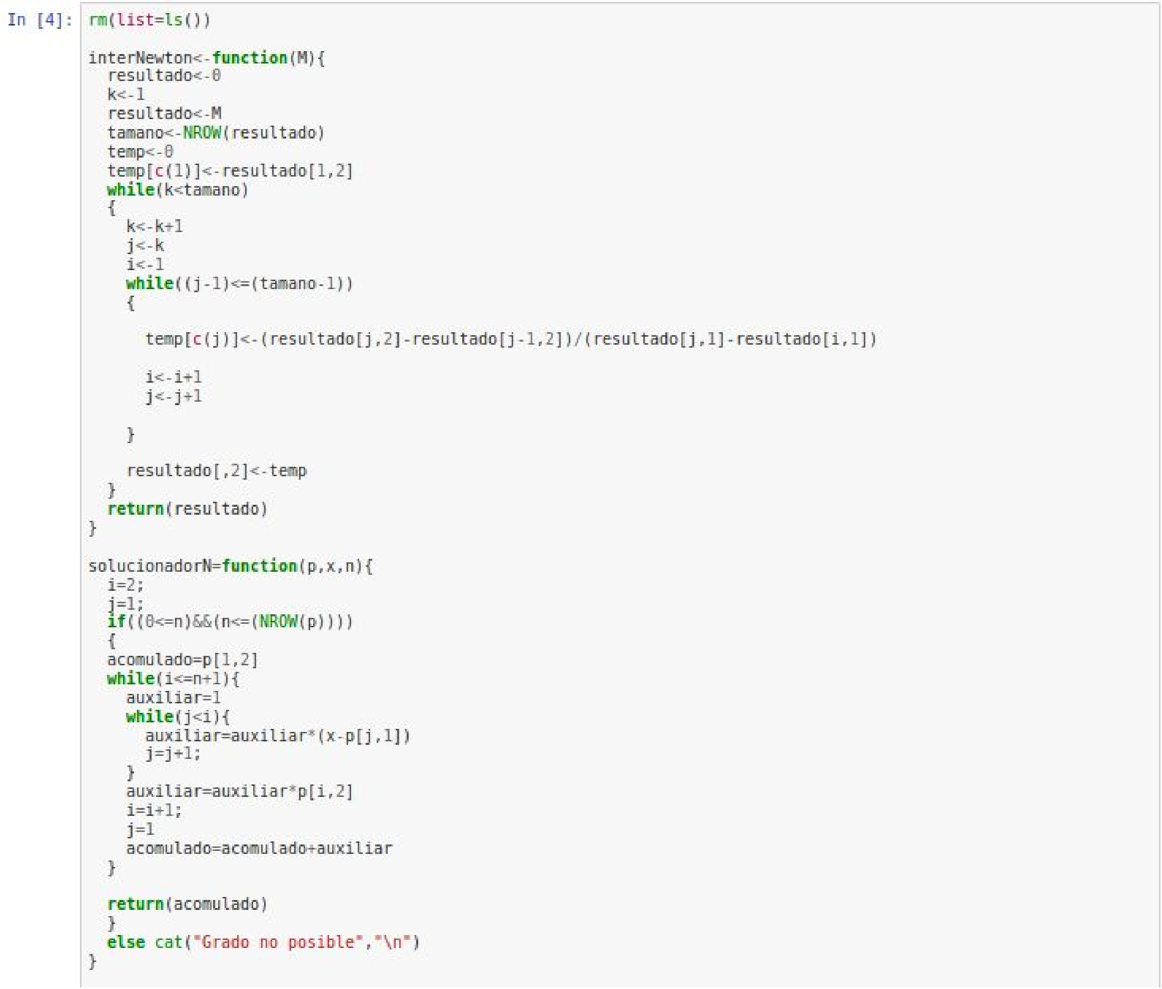
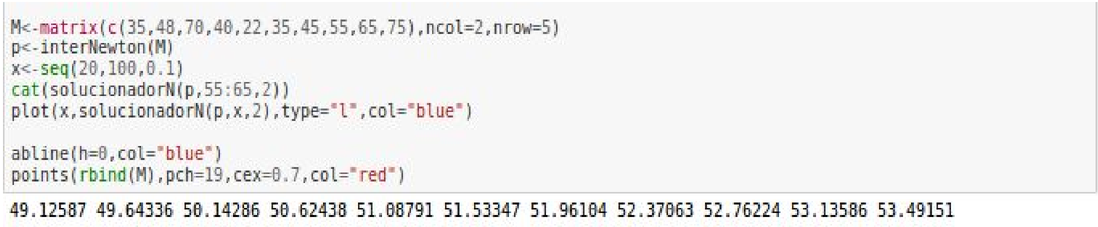
Parcial 2
Luis Fernando Rodriguez

Punto 1.

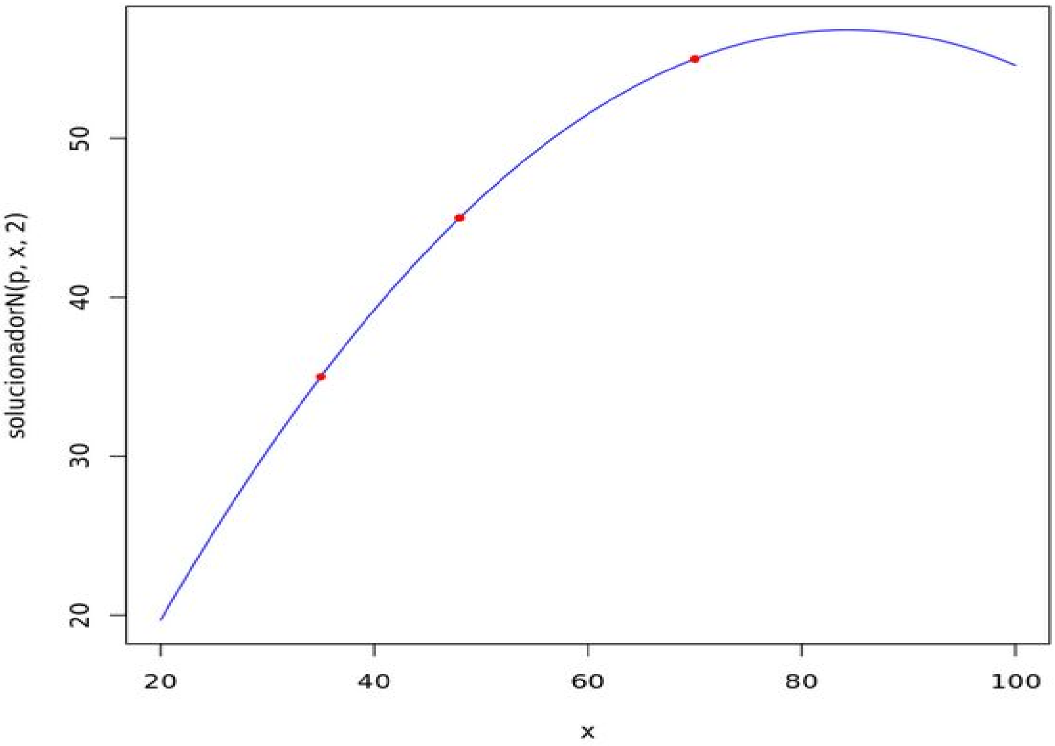


```
In [4]: rm(list=ls())

        interNewton<-function(M){
          resultado<-0
          k<-1
          resultado<-M
          tamano<-NROW(resultado)
          temp<-0
          temp[c(1)]<-resultado[1,2]
          while(k<tamano)
          {
            k<-k+1
            j<-k
            i<-1
            while((j-1)<=(tamano-1))
            {

              temp[c(j)]<-(resultado[j,2]-resultado[j-1,2])/(resultado[j,1]-resultado[i,1])

              i<-i+1
              j<-j+1

            }

            resultado[,2]<-temp
          }
          return(resultado)
        }

        solucionadorN=function(p,x,n){
          i=2;
          j=1;
          if((0<=n)&&(n<=(NROW(p))))
          {
          acomulado=p[1,2]
          while(i<=n+1){
            auxiliar=1
            while(j<i){
              auxiliar=auxiliar*(x-p[j,1])
              j=j+1;
            }
            auxiliar=auxiliar*p[i,2]
            i=i+1;
            j=1
            acomulado=acomulado+auxiliar
          }

          return(acomulado)
          }
          else cat("Grado no posible","\n")
        }
```



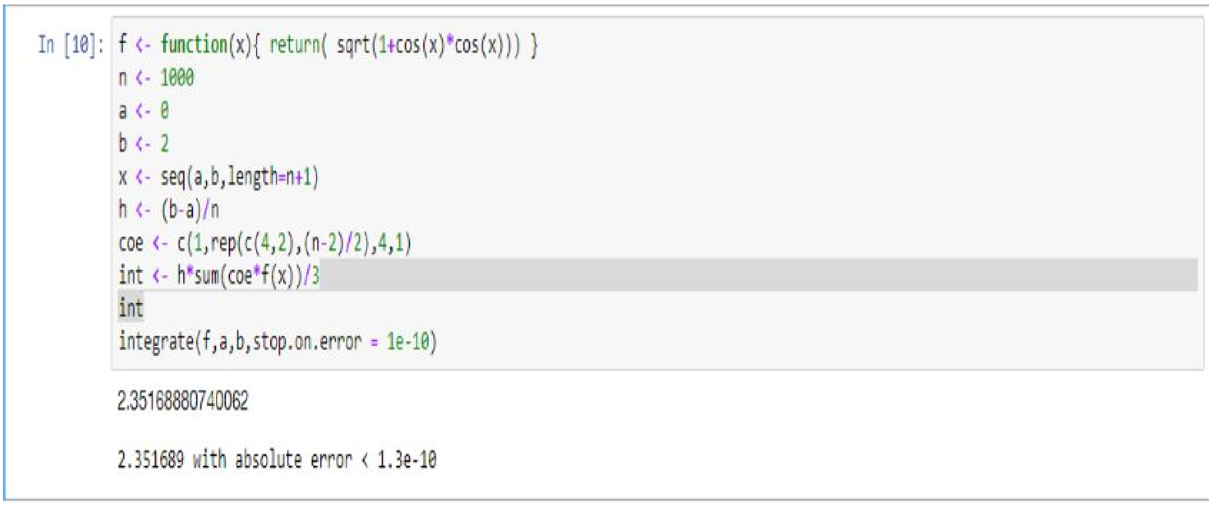
```
M<-matrix(c(35,48,70,40,22,35,45,55,65,75),ncol=2,nrow=5)
p<-interNewton(M)
x<-seq(20,100,0.1)
cat(solucionadorN(p,55:65,2))
plot(x,solucionadorN(p,x,2),type="l",col="blue")

abline(h=0,col="blue")
points(rbind(M),pch=19,cex=0.7,col="red")
```

```
49.12587 49.64336 50.14286 50.62438 51.08791 51.53347 51.96104 52.37063 52.76224 53.13586 53.49151
```



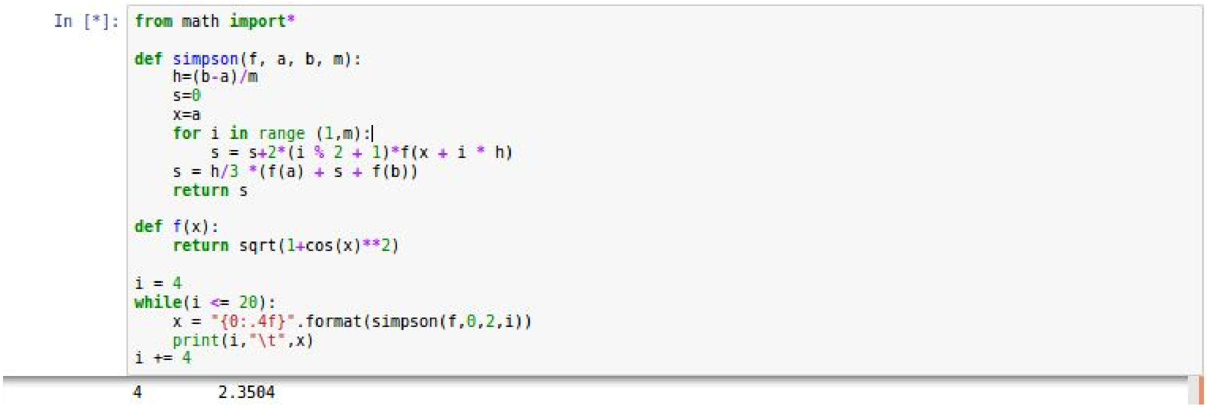
Punto 2
Regla de Simpson

Solucion en R



```
In [10]: f <- function(x){ return( sqrt(1+cos(x)*cos(x))) }
         n <- 1000
         a <- 0
         b <- 2
         x <- seq(a,b,length=n+1)
         h <- (b-a)/n
         coe <- c(1,rep(c(4,2),(n-2)/2),4,1)
         int <- h*sum(coe*f(x))/3
         int
         integrate(f,a,b,stop.on.error = 1e-10)

         2.35168880740062

         2.351689 with absolute error < 1.3e-10
```

Solución en Phyton



In [*]:

```python
from math import*

def simpson(f, a, b, m):
    h=(b-a)/m
    s=0
    x=a
    for i in range (1,m):
        s = s+2*(i % 2 + 1)*f(x + i * h)
    s = h/3 *(f(a) + s + f(b))
    return s

def f(x):
    return sqrt(1+cos(x)**2)

i = 4
while(i <= 20):
    x = "{0:.4f}".format(simpson(f,0,2,i))
    print(i,"\t",x)
i += 4
```

```
4        2.3504
```