Computing Group 5 HW 6

November 22, 2020

1 126.

```
cos_fixed = function(x, tol, i=0){
  while(signif(x, tol) != signif(cos(x), tol)){
    x=cos(x)
    i=i+1
  }
  return(c(x, cos(x), i))
}

# (x, cos(x), number of iterations)
cos_fixed(0.5, 2)

## [1]  0.7350063  0.7418265 10.0000000

cos_fixed(0.5, 3)

## [1]  0.7387045  0.7393415 16.0000000

cos_fixed(0.5, 4)

## [1]  0.7391091  0.7390690 23.0000000

cos_fixed(0.7, 2)

## [1] 0.7444212 0.7354802 5.0000000

cos_fixed(0.7, 3)

## [1]  0.7387487  0.7393117 12.0000000

cos_fixed(0.7, 4)

## [1]  0.7391318  0.7390537 17.0000000

cos_fixed(0, 2)
```

```
## [1]  0.7442374  0.7356047 11.0000000
```

```
cos_fixed(0, 3)
```

```
## [1]  0.7387603  0.7393039 18.0000000
```

```
cos_fixed(0, 4)
```

```
## [1]  0.7391302  0.7390548 23.0000000
```

Foo

2 132.

```
hornerpoly <- function(x,a) {
  res <- numeric(length(x))
  for(j in 1:length(x)) {
    v <- a
    for(i in (length(a)-1):1) {
      v[i] <- (v[i+1]*x[j]) + a[i]
    }
    res[j]<-v[1]
  }
  return(res)
}

polyderiv=function(beta){
  beta_deriv=beta*(seq(1:length(beta))-1)
  return(beta_deriv[-1])
}

newton_poly = function(x, beta, eps){
  counter = 0
  while(abs(hornerpoly(x, beta))>eps){
    x = x-hornerpoly(x, beta)/hornerpoly(x, polyderiv(beta))
    counter = counter + 1
  }
  return(c(x, counter))
}

newton_poly(0, c(5, 0.0005, 605, 0.0605, 10600, 1.06, 10000, 1), eps=0.0001)
```

```
## [1] -10000      1
```

```
#The solution is found after just 1 iteration.
```

3 136.

To Do

4 137.

$f(x) = 1/x - y \Rightarrow f'(x) = -1/x^2 \Rightarrow \frac{f(x)}{f'(x)} = \frac{1/x-y}{-1/x^2} = -(1/x - y)x^2 = yx^2 - x$
$\Rightarrow x_{n+1} = x_n - (yx_n^2 - x_n) = 2x_n - yx_n^2$

Applying this method iteratively will find the reciprocal of y.