R code for question 1

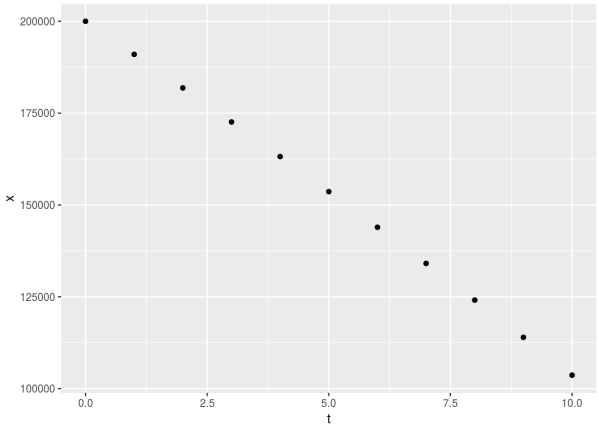
```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.2.1 ──
```

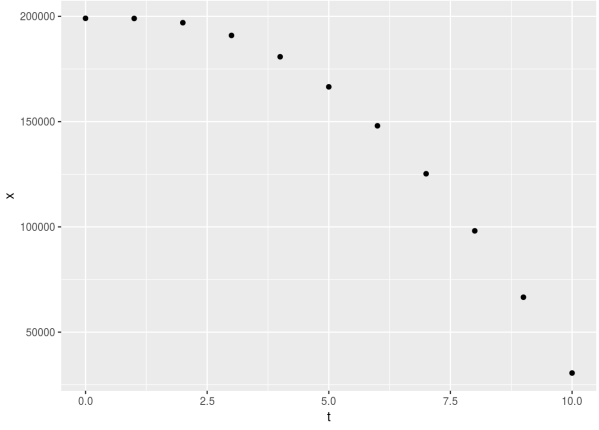
```
## ✔ ggplot2 2.2.1     ✔ purrr   0.2.4
## ✔ tibble  1.4.2     ✔ dplyr   0.7.4
## ✔ tidyr   0.8.0     ✔ stringr 1.3.0
## ✔ readr   1.1.1     ✔ forcats 0.3.0
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
make_constant <- function(r,u,x0) {
  k <- x0 - u/r
  Vectorize(function(t) {
    k*(1+r)**t + u/r
  })
}

data_frame(t = 0:10) %>%
  mutate(x = make_constant(0.015, 12000, 200000)(t)) %>%
  ggplot(aes(t, x)) +
  geom_point()
```



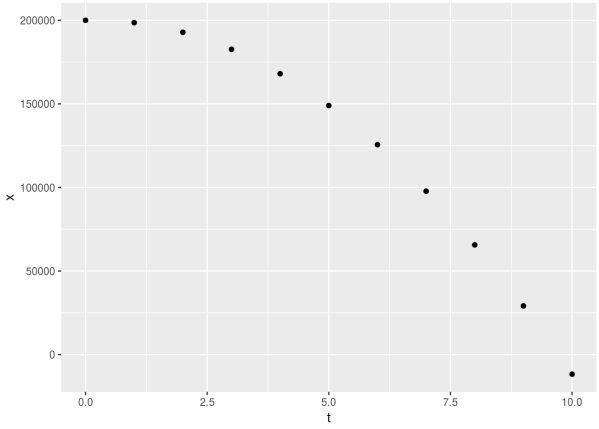

```
make_linout <- function(r,u,alp,x0) {
  out <- function(tau) {
    u+alp*tau
  }
  summand <- function(tau) {
    (-out(tau))/((1+r)**tau)
  }
  Vectorize(function(t) {
    s <- sum(map_dbl(0:(t-1), summand))
    ((1+r)**(t-1))*(x0 + s)
  })
}

data_frame(t = 0:10) %>%
  mutate(x = make_linout(0.015, 1000, 4000, 200000)(t)) %>%
  ggplot(aes(t, x)) +
  geom_point()
```




```
uniroot(function(alp) {make_linout(0.015, 1000, alp, 200000)(10)},
        lower=4000, upper=5000)$root
```

```
## [1] 4652.808
```

```
make_decr_rent <- function(r0, r, u, alpha) {
  function(xt, t) {
    (1 + r0*(r**t))*xt - (u + alpha*t)
  }
}

decr_rent <- make_decr_rent(0.02,0.9,1000,4000)
xs <- vector(length = 11)
xs[1] <- 200000
for (i in 1:10) {
  xs[i+1] <- decr_rent(xs[i],i)
}

data_frame(t = 0:10, x = xs) %>%
  ggplot(aes(t, x)) +
  geom_point()
```

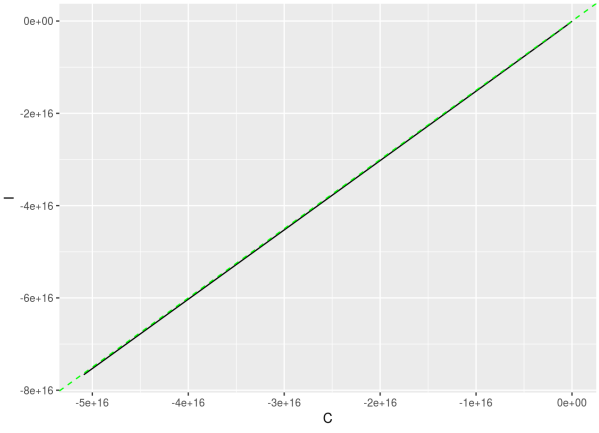


R code for question 2

```
a <- 0.8
b <- 10000
c <- 3
c(-a*(c+1), 1+a*c)
```

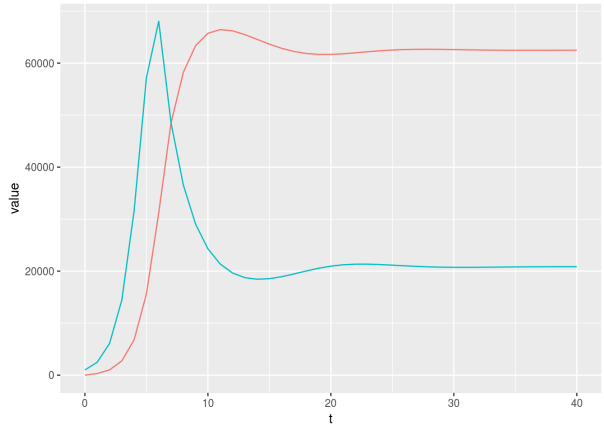
```
## [1] -3.2  3.4
```

```
A <- matrix(c(a, a, (a-1)*c, a*c), nrow=2, byrow = T)
add <- c(b, b*c)

move <- function(now) {
  A%*%now + add
}

evolve <- function(x0, n) {
  xs <- matrix(ncol = 2, nrow = n + 1)
  xs[1,] <- x0
  for (i in (2:(n+1))) {
    xs[i,] <- move(xs[i-1,])
  }
  xs <- data.frame(xs)
  colnames(xs) <- c('C', 'I')
  round(xs, 2)
}
evolve(c(1000000, -50000000000000), 10) %>%
  ggplot(aes(C,I)) +
  geom_path() +
  geom_abline(intercept = 0,
              slope = 3/2,
              color = 'green',
              linetype='dashed')
```

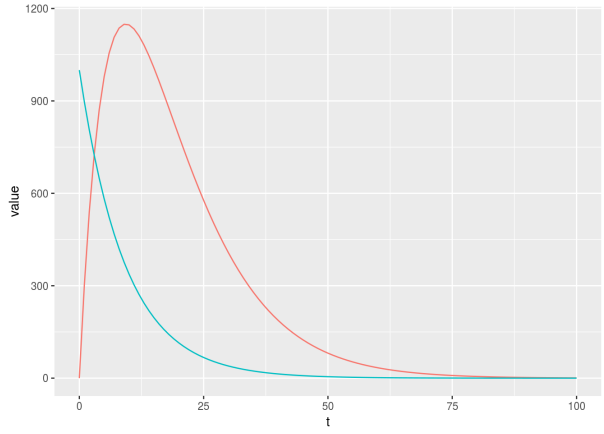


R code for question 3

```
make_movement <- function(a,b,c,N) {
  function(now) {
    st <- now[1]
    it <- now[2]
    nexts <- (1-a)*st + b*st*(1 - (st+it)/N)
    nexti <- (1-c)*it + a*st
    c(nexts,nexti)
  }
}

evolve <- function(move, x0, n) {
  xs <- matrix(ncol = 2, nrow = n + 1)
  xs[1,] <- x0
  for (i in (2:(n+1))) {
    xs[i,] <- move(xs[i-1,])
  }
  xs <- data.frame(xs)
  colnames(xs) <- c('S', 'I')
  xs$t <- 0:n
  round(xs, 2)
}

N <- 100000
a <- 0.3
b <- 1.8
c <- 0.1

evolve(make_movement(a, b, c, N),
       c(1000,0),
       40) %>%
  gather(key, value, S, I) %>%
  ggplot(aes(t, value, color=key)) +
  geom_line() +
  theme(legend.position = 'none')
```




```
b <- 0.2

evolve(make_movement(a, b, c, N),
       c(1000,0),
       100) %>%
  gather(key, value, S, I) %>%
  ggplot(aes(t, value, color=key)) +
  geom_line() +
  theme(legend.position = 'none')
```



```
gamma <- function(a,b,c,N) {
  N*(b-a)/(b*(a+c))
}

b <- 1.8

Sstar <- c*gamma(a, b, c, N)
Istar <- a*gamma(a, b, c, N)

funcmat <- function(a, b, c, N, Sstar, Istar) {
 A11 <- 1-a+b-b*((2*Sstar + Istar)/N)
 A12 <- -b*Sstar/N
 A21 <- 1-c
 A22 <- a
 matrix(c(A11,A21,A12,A22), nrow=2)
}

A1 <- funcmat(a, b, c, N, Sstar, Istar)
A2 <- funcmat(a, b, c, N, 0, 0)
A1
```

```
##       [,1]   [,2]
## [1,] 0.625 -0.375
## [2,] 0.900  0.300
```

```
A2
```

```
##     [,1] [,2]
## [1,]  2.5  0.0
## [2,]  0.9  0.3
```

```
eigen(A1)
```

```
## eigen() decomposition
## $values
## [1] 0.4625+0.5577578i 0.4625-0.5577578i
##
## $vectors
##                       [,1]                [,2]
## [1,] 0.1516970+0.5206781i 0.1516970-0.5206781i
## [2,] 0.8401681+0.0000000i 0.8401681+0.0000000i
```

```
eigen(A2)
```

```
## eigen() decomposition
## $values
## [1] 2.5 0.3
##
## $vectors
##           [,1] [,2]
## [1,] 0.9255470    0
## [2,] 0.3786328    1
```

```
Mod(eigen(A1)$values)
```

```
## [1] 0.7245688 0.7245688
```