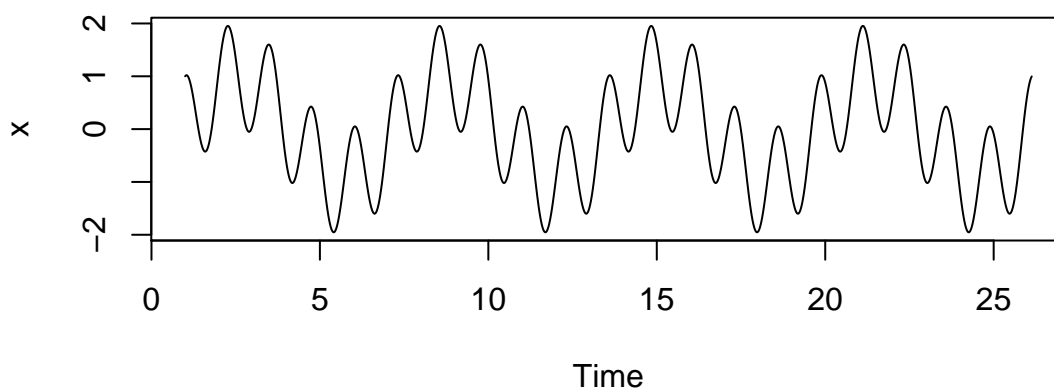


```
library("quantmod")

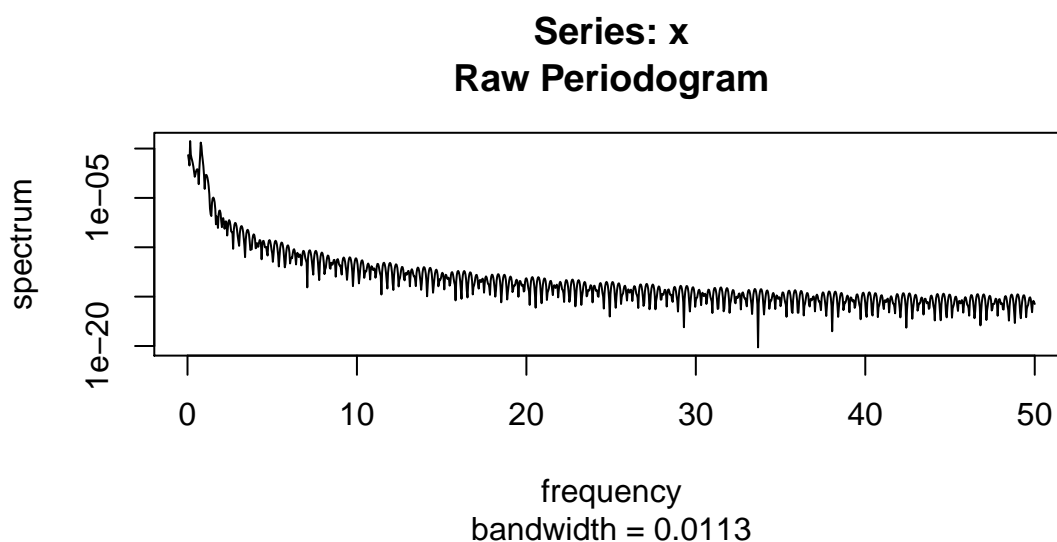
# gestosc spektralna sluzzy do wykrywania istotnych czestotliwosci
# (czyli pewnej sezonowosci)
#  $T = 1/f$  (gdzie  $T$  to okres)
# wiec jesli pik w gestosci jest w omega, to okres przyjmujemy  $1/\omega$ 

# zad.1

x <- sin(seq(0,8*pi,0.01)) + cos(5*seq(0,8*pi,0.01))
# ta funkcja ma okres  $2/5\pi$ 
x <- ts(x,frequency=100)
plot(x,type="l")
```



```
S <- spectrum(x,ci=0) # rysuje periodogram
```



```
S$freq[which.max(S$spec)] # czestotliwosc, dla ktorej osiagane

## [1] 0.1562

# jest maksimum gestosci
1/(S$freq[order(-S$spec)[1:2]]) # interesuja nas okresy, czyli jeden nad

## [1] 6.40 1.28
```

```

# czestotliwosc, szukamy dwoch
# najwiekszych wartosci
# dwa istotne okresy to to powyzej

c(2*pi,2/5*pi) # a teoretycznie powinniśmy dostac to

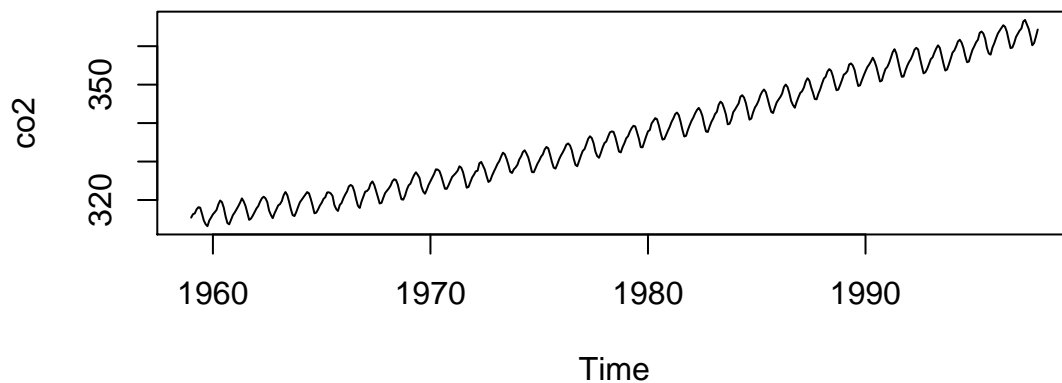
## [1] 6.283 1.257

# periodogram: dlaczego na osi x jest od zera do 50?
# odp: połowa czestotliwosci, które braliśmy
# widzimy tu dwa maksima

# zad.2

data(co2)
plot(co2)

```

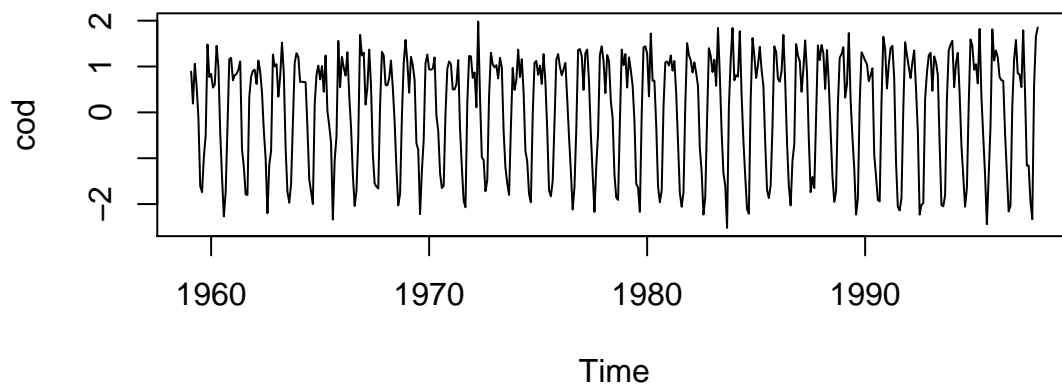


```

# szereg z trendem i okresowy

cod <- diff(co2)
plot(cod) # pozbyliśmy się trendu, a okresowosc została

```

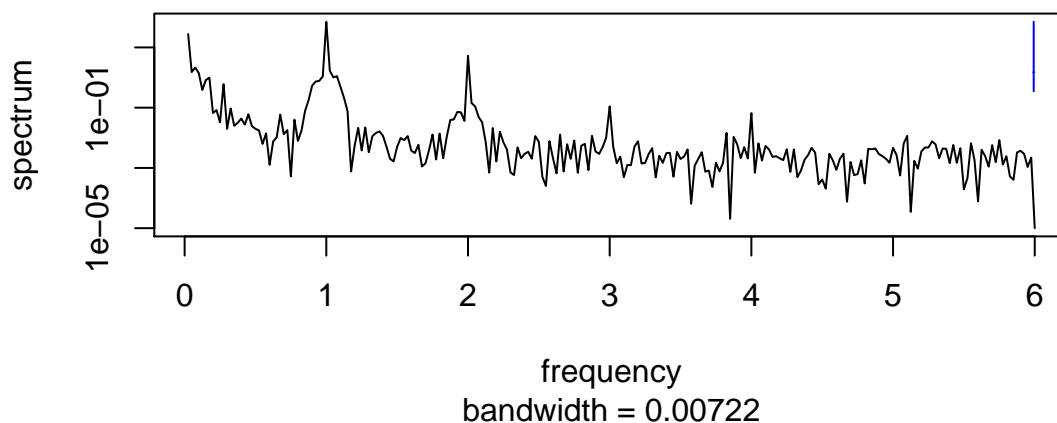


```

sp <- spectrum(co2) # maksimum w jedynce i dwojce

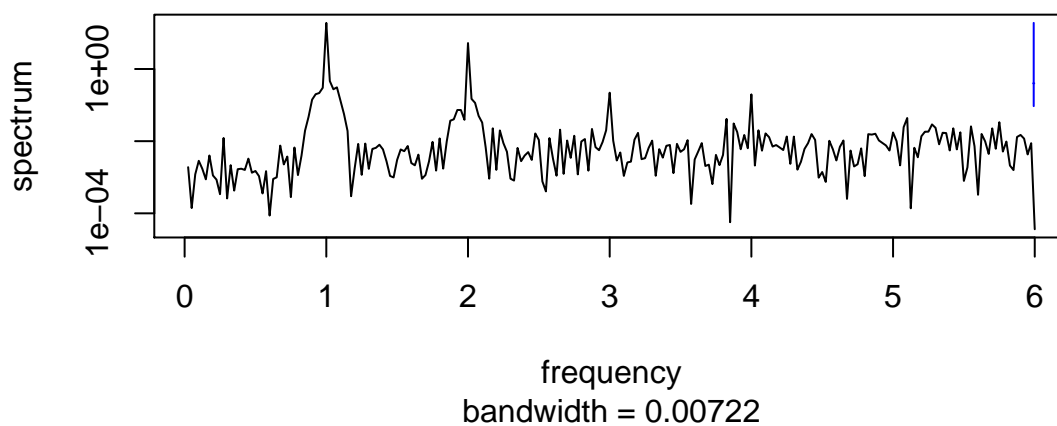
```

Series: x Raw Periodogram



```
sp.diff <- spectrum(cod) # maksima zostaja, ale pozbylismy sie trendu, który moglby nam przeszkodzić przy es
```

Series: x Raw Periodogram

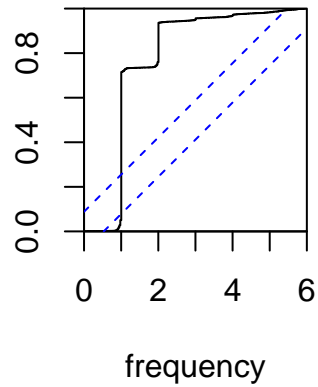


```
# ta niebieska kreska to przedział ufności, niekoniecznie symetryczny,  
# tam gdzie kropka to środek
```

```
# jak to interpretować? maksimum równe 1? to okres 1/1, czyli 1, ale czego?  
# jakiej jednostki? 1 rok! dlatego kończy się na 6, bo wtedy okres to 1/6,  
# czyli dwa miesiące, a częściej już się nie da
```

```
cpgram(cod)
```

Series: cod



```
# dystrybuanta spektralna (dwa skoki, czyli mamy dwie składowe okresowe
# (najprawdopodobniej)
# pasek na srodku to pasy ufnosci dla hipotezy o białym szumie
# (czyli sprawdzamy, czy jest to biały szum: tak, jeśli cały wykres mieści
# się w niebieskich paskach)

# istotne częstotliwości (w latach):
sp.diff$freq[order(-sp.diff$spec)[1:2]]

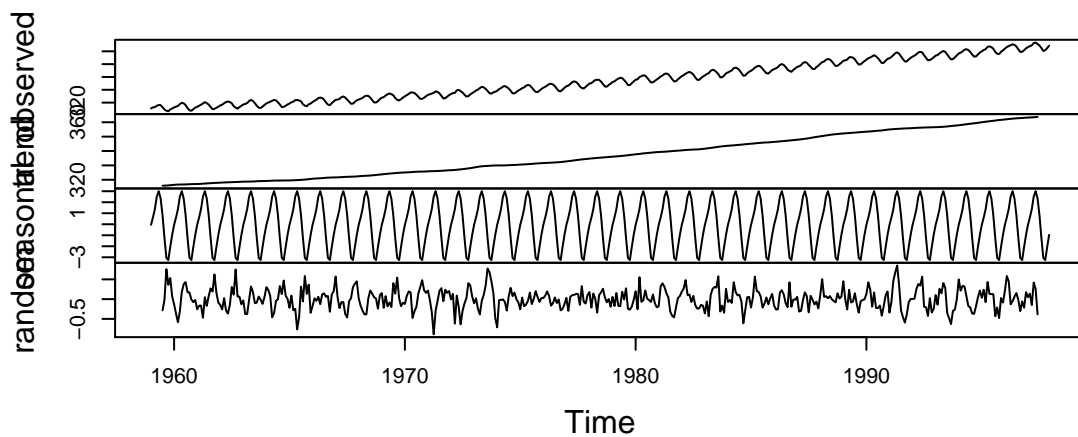
## [1] 1 2

# istotne okresy:
1/sp.diff$freq[order(-sp.diff$spec)[1:2]]

## [1] 1.0 0.5

d <- decompose(co2)
plot(d)
```

Decomposition of additive time series

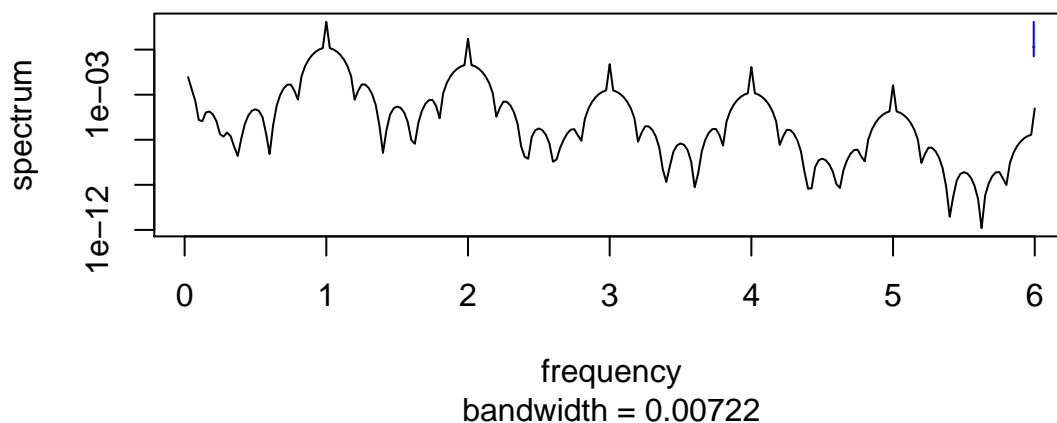


```
# robi dekompozycje na czesc sezonowa, trend, czesc losowa i takie tam

# zajmijmy sie składowa sezonowa:

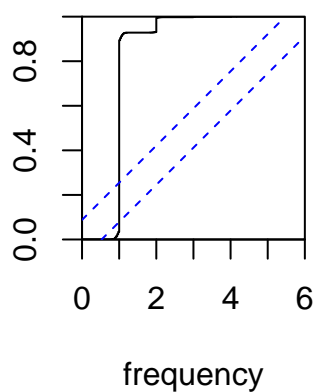
sp.sez <- spectrum(d$seasonal)
```

Series: x Raw Periodogram



```
cpgram(d$seasonal)
```

Series: d\$seasonal



```
sp.sez$freq[order(-sp.sez$spec)[1]] # czyli jeden rok (co 12 obserwacji)
```

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

```
# mamy trzy składowe okresowe -> chcemy znaleźć odpowiadające im
#                                     częstotliwości
```

```
# uwaga! szukamy maksimum lokalnego!
```

```
loc.max.coor <- function(x) {
  n <- length(x)
  Znak <- diff(x)*Lag(diff(x))
  ex_indices <- c(1,which(Znak<0 & diff(x)<0 ), n) # maksima + brzegi
  return (ex_indices)
}
```

```
loc.max.coor(c(0,3,2,6,0,-1,9,10,1)) # maly test na tej funkcji ->
```

```
## [1] 1 2 4 8 9
```

```
# rzeczywiscie zwraca maksima i brzegi
```

```

ist.czest <- sp.diff$freq[loc.max.coor(sp.diff$spec)]
# istotne czestotliwosci

ist.czest[order(-sp.diff$spec[loc.max.coor(sp.diff$spec)])]

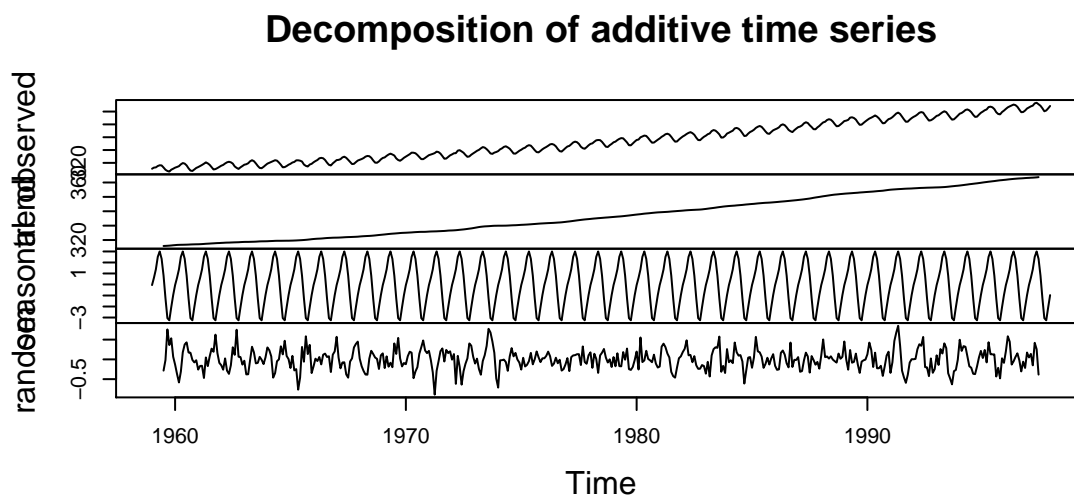
## [1] 1.000 2.000 1.075 3.000 4.000 1.925 5.100 3.825 5.750 3.875 5.275
## [12] 5.550 2.175 5.425 5.700 2.875 2.650 4.050 2.225 5.475 5.000 5.350
## [23] 3.200 4.100 2.475 4.875 5.625 4.425 4.825 5.900 3.950 2.750 4.250
## [34] 4.300 0.275 2.575 1.800 2.825 3.550 4.650 2.700 3.300 4.550 5.800
## [45] 1.750 3.650 5.975 1.275 3.475 1.225 1.375 3.425 4.175 0.675 1.575
## [56] 0.775 5.150 2.350 4.775 3.350 1.525 4.700 2.425 3.075 1.650 3.750
## [67] 0.175 0.725 0.450 0.100 3.700 0.325 0.025 0.400 0.500 0.575 4.500
## [78] 6.000

# posortowane istotne czestotliwosci

# jeszcze raz:

plot(d)

```



```

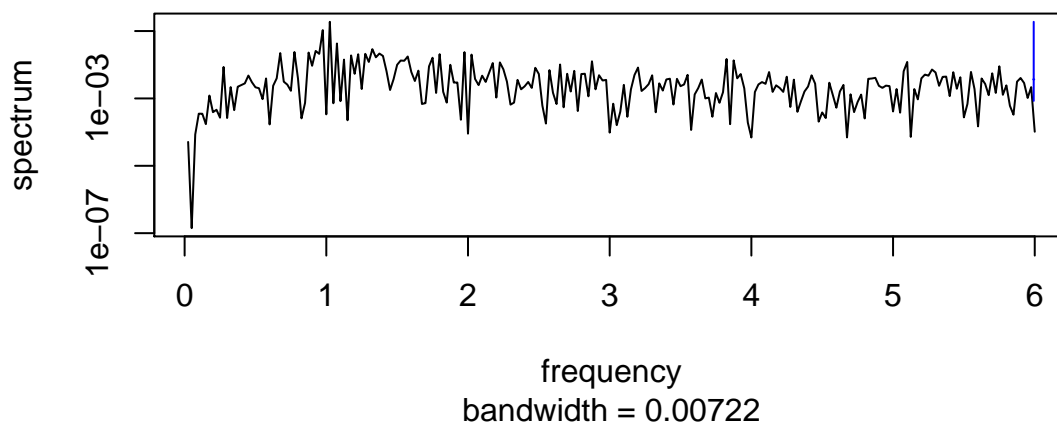
# czemu sa braki danych? bo byla srednia ruchoma, wiec czesc danych tracimy
# dlaczego czesc sezonowa jest dluzsza mimo, ze braki danych? bo obserwacje
# sa doliczane zgodnie z sezonowoscia

# zajmiemy sie czescia losowa:

sp.rand <- spectrum(window(d$random, start=c(1959,7), end=c(1997,6)))

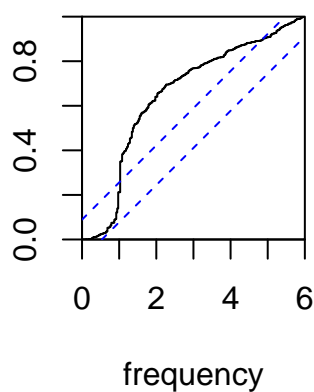
```

Series: x Raw Periodogram



```
cpgram(d$random) # to nie jest biały szum, a raczej powinien być
```

Series: d\$random



```
# ale to jeszcze nie koniec świata, być może możemy  
# dopasować do tego jakiś model arma
```

```
# predykcja kolejnych 50 elementów:
```

```
m <- HoltWinters(co2, seasonal="additive")  
m$coefficients
```

```
##          a          b          s1          s2          s3          s4          s5          s6  
## 364.7616  0.1247  0.2215  0.9553  1.5985  2.8758  3.2820  2.4407  
##          s7          s8          s9          s10         s11         s12  
##  0.8969 -1.3796 -3.4112 -3.2570 -1.9135 -0.5844
```

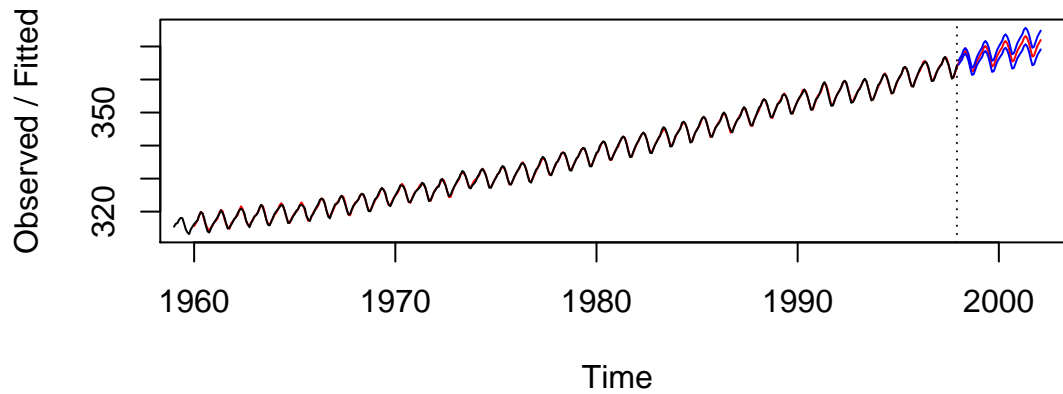
```
#  $X_{12k+i} = a + b(12k+2) + s_i$ 
```

```
# s odpowiada za sezonowość, zaś a i b za trend
```

```
# predykcja:
```

```
p <- predict(m, n.ahead=50, prediction.interval=TRUE)  
plot(m, p) # ładnie :D
```

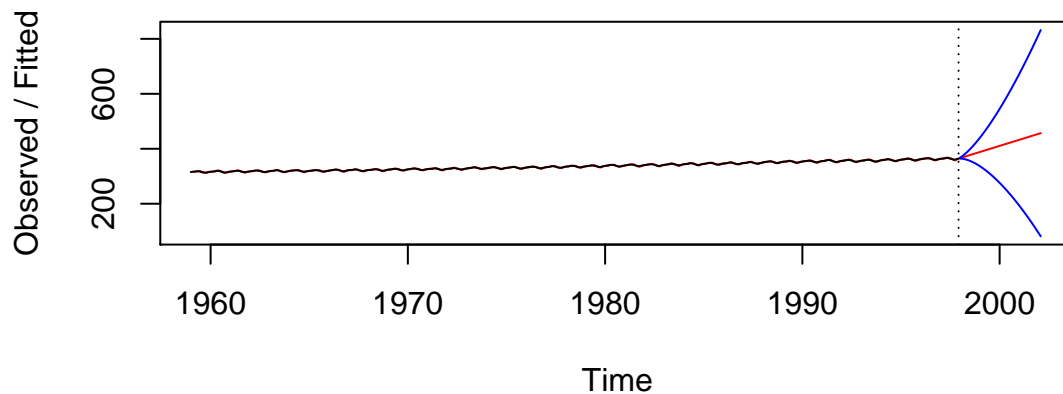
Holt–Winters filtering



```
# a teraz bez czesci sezonowej (parametr gamma za to odpowiada):
```

```
m <- HoltWinters(co2,seasonal="additive",gamma=FALSE)
p <- predict(m,n.ahead=50,prediction.interval=TRUE)
plot(m,p) # tragiczna predykcja, czyli ta czesc sezonowa
```

Holt–Winters filtering



```
# byla rzeczywiscie potrzebna
```

```
# zad.3
```

```
n <- 60
eps <- rnorm(n)
x <- numeric(n)
y <- numeric(n)

# w przod:

x[1] <- 0
for(i in 2:n){
  x[i] <- 2*x[i-1] + eps[i]
}
```

```
# w tyl:
```

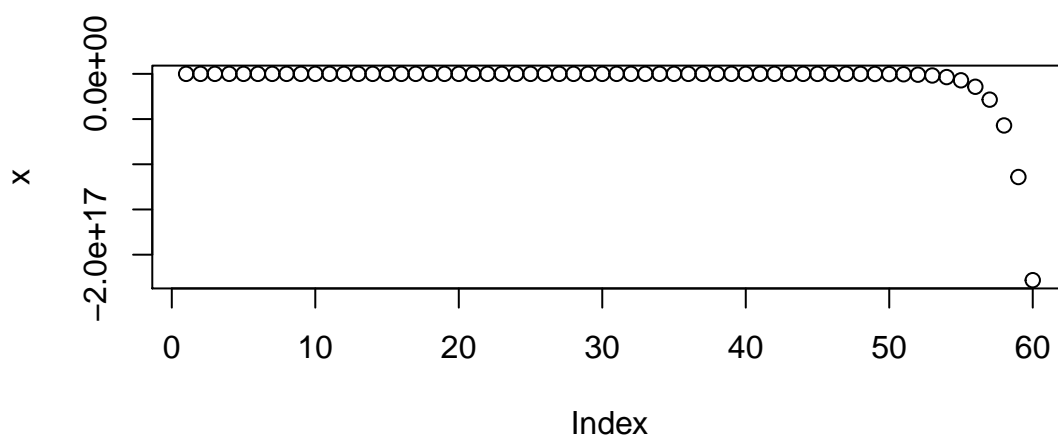


```

y[60] <- 0
for(i in n:2){
  y[i-1] <- (y[i]-eps[i])/2
}

plot(x)

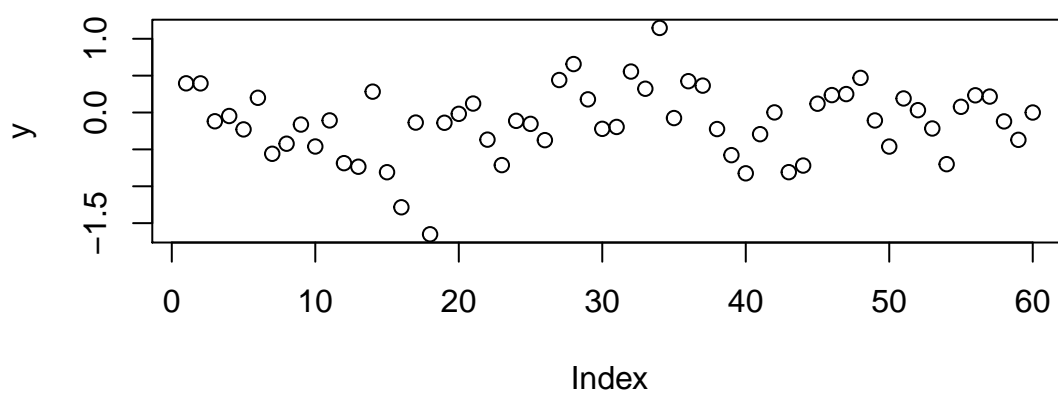
```



```

plot(y)

```



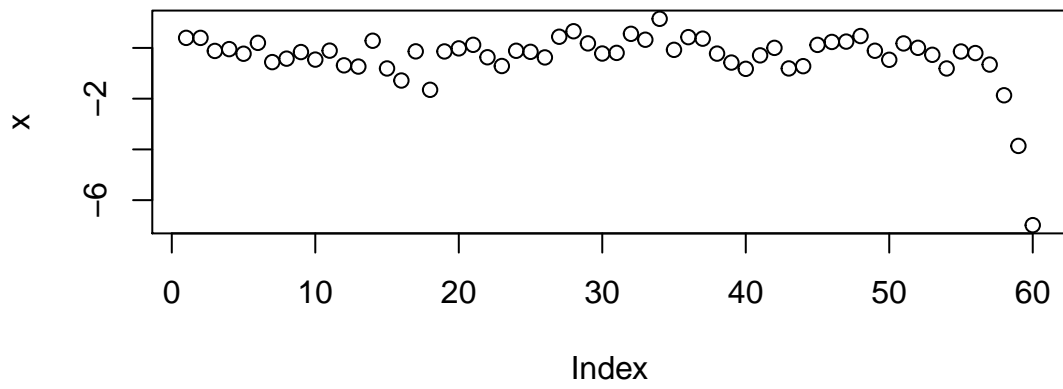
```

# r  une

x[1] <- y[1]
for(i in 2:n){
  x[i] <- 2*x[i-1] + eps[i]
}

plot(x)

```



```
# powinny byc sobie rowne, wiec skad sie bierze roznic?  
# odp: z bledow numerycznych
```