true

2021-01-23

Contents

.

KCD 2020 .

Chapter 1

Chapter 2

- (Prime Number): 1 , 1 ( )
- (Composite Number):

  − 1
- (Prime Factor):
- (Coprime, Relatively Prime): 1

2.1

, N . N p, q .
N p q . N .(,
2020)

2.1.1

Prime number function in R . N $2 \sim N-1$
. %% . , 10 3 1 . 10
%% 3 1 .

```
is_prime <- function(num) {
   if (num == 2) {
      TRUE
   } else if (any(num %% 2:(num-1) == 0)) {
      FALSE
   } else {
```

```
      TRUE
   }
}

is_prime(3)
```

```
[1] TRUE
```

$1 \sim 100$.

```
library(tidyverse)

natural_number <- 1:100

prime_number_decision <- map_lgl(natural_number, is_prime)

tibble(     = natural_number,
           = prime_number_decision) %>%
  reactable::reactable()
```

자연수	소수판정
1	false
2	true
3	true
4	false
5	true
6	false
7	true
8	false
9	false
10	false

1–10 of 100 rows Previous 1 2 3 4 5 ... 10 Next

N N . reactable .

```
library(reactable)
calculate_primes <- function(number) {

  natural_number <- 1:number

  prime_number_decision <- map_lgl(natural_number, is_prime)

  tibble(     = natural_number,
              = prime_number_decision) %>%
    reactable::reactable(columns = list(
            = colDef(
          cell = function(value) {
            if (value == TRUE ) paste0("", value) else value
          },
```

```
          style = function(value) {
            color <- if (value == TRUE) {
              "#008000"
            } else{
              "#e00000"
            }
            list(fontWeight = 600, color = color)
          }
        )
      ))
}

calculate_primes(10)
```

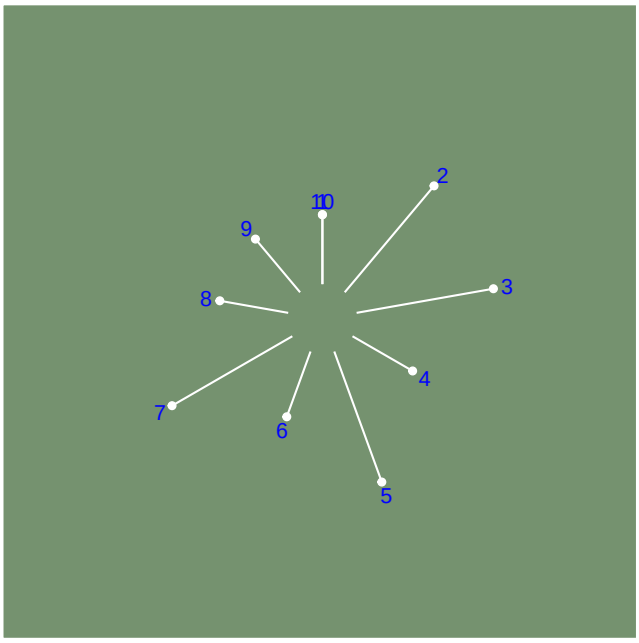
자연수	소수판정
1	FALSE
2	TRUE
3	TRUE
4	FALSE
5	TRUE
6	FALSE
7	TRUE
8	FALSE
9	FALSE
10	FALSE

2.1.2

.

```
##
generate_primes <- function(number) {

  natural_number <- 1:number
  prime_number_decision <- map_lgl(natural_number, is_prime)

  decide_prime_tbl <- tibble(     = natural_number,
                                  = prime_number_decision %>% as.integer + 1)

  return(decide_prime_tbl)
}

prime_tbl <- generate_primes(10)

##
lines <- tibble(number = prime_tbl %>% pull(`  `),
                x      = seq(1, 10, by = 1),
                xend   = x,
                y      = rep(0, 10),
                yend   = prime_tbl %>% pull(`  `))

dots <- lines %>%
  select(x, yend)

##    ----
lines %>% ggplot() +
  geom_segment(aes(x = x, xend = xend,
                   y = y, yend = yend),
               color = "white") +
  geom_text(aes(x = x, y = yend + 0.2, label = number),
            color = "blue") +
  geom_point(data = dots,
             aes(x = x, y = yend),
             color = "white") +
  coord_polar() +
  ylim(-0.5, 3) +
  coord_polar() +
  theme(
    plot.background = element_rect(
      fill = "#75926f"),
    panel.background = element_rect(
      fill = "#75926f"),
```

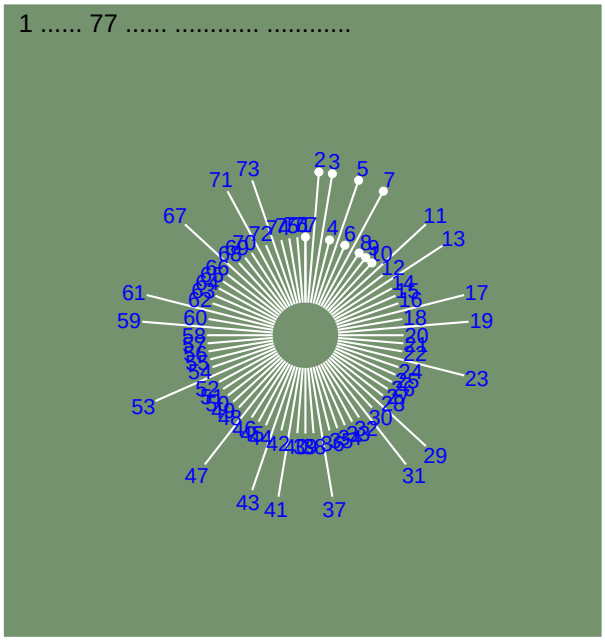
```
    panel.grid = element_blank(),
    plot.caption = element_text(
      family = "Open Sans",
      size = 6,
      color = "white"),
    axis.title = element_blank(),
    axis.text = element_blank(),
    axis.ticks = element_blank()
  )
```



. , .

```
generate_primes <- function(number) {

  natural_number <- 1:number
  prime_number_decision <- map_lgl(natural_number, is_prime)

  decide_prime_tbl <- tibble(     = natural_number,
                                  = prime_number_decision %>% as.integer + 1)
  return(decide_prime_tbl)
}

visualize_prime <- function(number) {
```

```
  #

  prime_tbl <- generate_primes(number)

  graph_tbl <- tibble( natural_number = prime_tbl %>% pull(`  `),
                       x              = seq(1, number, by = 1),
                       xend           = x,
                       y              = rep(0, number),
                       yend           = prime_tbl %>% pull(`  `)
  )

  dots <- lines %>%
    select(x, yend)

  ##    ----
  graph_tbl %>% ggplot() +
    geom_segment(aes(x = x, xend = xend,
                     y = y, yend = yend),
                 color = "white") +
    geom_text(aes(x = x, y = yend + 0.2, label = natural_number),
              color = "blue") +
    geom_point(data = dots,
               aes(x = x, y = yend),
               color = "white") +
    coord_polar() +
    ylim(-0.5, 3) +
    coord_polar() +
    theme(
      plot.background = element_rect(
        fill = "#75926f"),
      panel.background = element_rect(
        fill = "#75926f"),
      panel.grid = element_blank(),
      plot.caption = element_text(
        family = "Open Sans",
        size = 6,
        color = "white"),
      axis.title = element_blank(),
      axis.text = element_blank(),
      axis.ticks = element_blank()
    ) +
    labs(title = glue::glue("1   {number}        "))
}

visualize_prime(77)
```

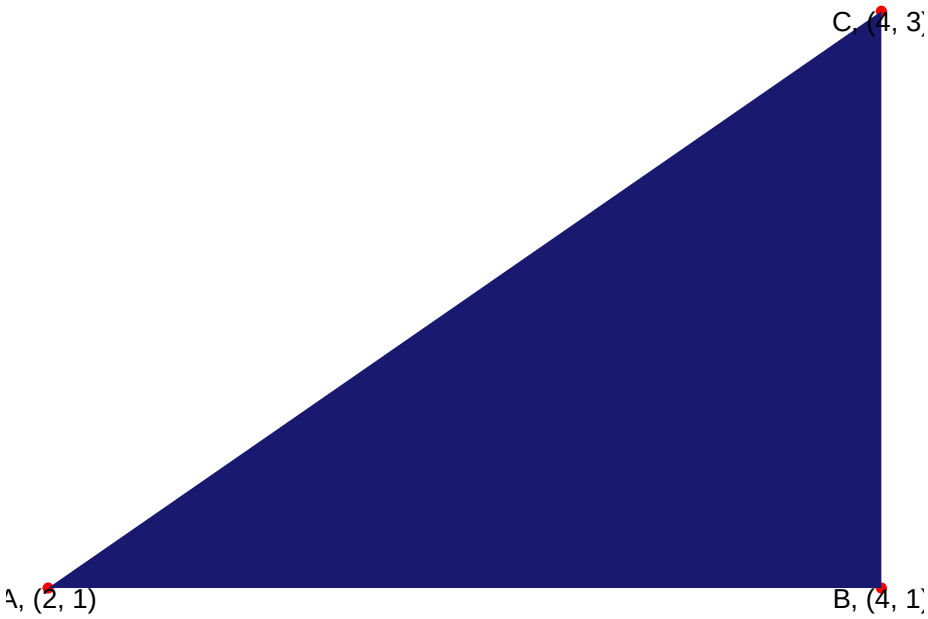

1 ..... 77 ....

Chapter 3

3.1

. ggplot . geom_polygon()
point geom_text() .

```
library(tidyverse)
triangle_tbl <- tibble(coord_x   = c(2,4,4),
                       coord_y   = c(1,1,3),
                       group     = c(1,1,1),
                       point     = c("A", "B", "C")) %>%
  mutate(point = glue::glue("{point}, ({coord_x}, {coord_y})"))

triangle_tbl %>%
  ggplot(aes(x     = coord_x,
             y     = coord_y,
             group = group,
             label = point)) +
    geom_point(size = 2, color = "red") +
    geom_polygon(fill ="midnightblue") +
    geom_text(vjust = 1, hjust= 0.5, size = 5) +
    theme_void()
```



3.2

ggplot 4 geom_rect() .

```
rect_tbl <- tibble(bottom_left  = 1,
                   bottom_right = 3,
                   top_left     = 1,
                   top_right    = 3)


rect_point_tbl <- rect_tbl %>%
  select(contains("left")) %>%
  rename(x=bottom_left, y=top_left) %>%
  bind_rows(
    rect_tbl %>%
      select(contains("right")) %>%
      rename(x=bottom_right, y=top_right)
  ) %>%
  mutate(point = c("A", "D")) %>%
  mutate(point = glue::glue("{point}:({x},{y})"))

ggplot() +
    geom_rect(data = rect_tbl, aes(xmin=bottom_left, xmax=bottom_right, ymin=top_left,
```

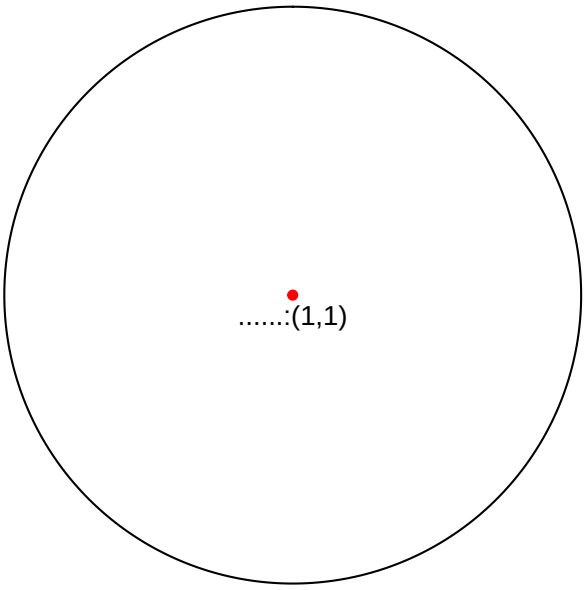
```
    geom_point(data = rect_point_tbl, mapping=aes(x=x, y=y), size = 2, color = "red") +
    geom_polygon(fill ="midnightblue") +
    geom_text(data = rect_point_tbl, mapping=aes(x=x, y=y, label = point), vjust = 1, hjust= 0.5,
    theme_void() +
    theme(legend.position = "none")
```

3.3 (circle)

ggforce geom_circle() (x, y) r .

```
library(ggforce)

circle_tbl <- tibble(x_pos = 1,
                     y_pos = 1) %>%
  mutate(center = glue::glue(" :({x_pos},{y_pos})"))

circle_tbl %>%
  ggplot(aes(x_pos, y_pos)) +
  geom_circle(aes(x0 = 1, y0 = 1, r = 1), inherit.aes = FALSE) +
  geom_point(size = 2, color = "red") +
  geom_text(aes(label = center), vjust = 1.5, hjust= 0.5, size = 5) +
  coord_fixed(ratio = 1) +
```

```
  theme_void() +
  theme(legend.position = "none")
```

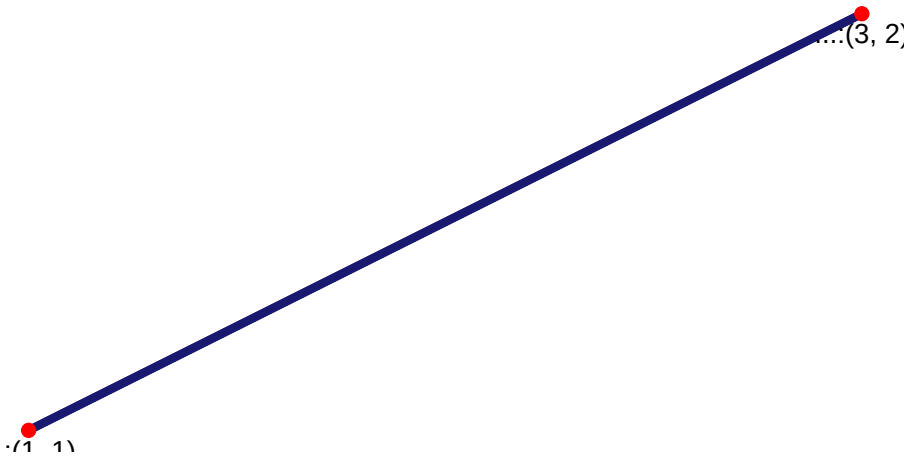


3.4 (line)

geom_segment() ggplot .

```
line_tbl <- tibble(x = c(1, 3),
                   y = c(1, 2)) %>%
  mutate(point = glue::glue(" :({x}, {y})"))

line_tbl %>%
  ggplot(aes(x, y, label = point)) +
    geom_segment(aes(x = line_tbl %>% select(x) %>% slice(1) %>% pull,
                     y = line_tbl %>% select(y) %>% slice(1) %>% pull,
                     xend = line_tbl %>% select(x) %>% slice(2) %>% pull,
                     yend = line_tbl %>% select(y) %>% slice(2) %>% pull),
                 size = 2,
                 linetype = 1,
                 color = "midnightblue") +
    geom_point(size = 3, color = "red")   +
    geom_text(vjust = 1.5, hjust= 0.5, size = 5) +
```

```
    coord_fixed(ratio = 1) +
    theme_void() +
    theme(legend.position = "none")
```



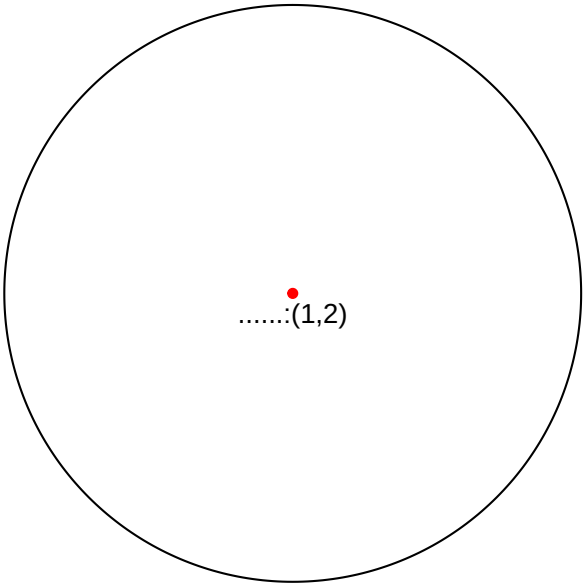
3.5

.

x, y r .

```
draw_circle <- function(x_pos, y_pos, radius) {

  circle_tbl <- tibble(x_pos = x_pos,
                       y_pos = y_pos) %>%
  mutate(center = glue::glue(" :({x_pos},{y_pos})"),
         radius = radius)

  circle_tbl %>%
    ggplot(aes(x_pos, y_pos)) +
    geom_circle(aes(x0 = x_pos, y0 = y_pos, r = radius), inherit.aes = FALSE) +
    geom_point(size = 2, color = "red") +
    geom_text(aes(label = center), vjust = 1.5, hjust= 0.5, size = 5) +
```

```
    coord_fixed(ratio = 1) +
    theme_void() +
    theme(legend.position = "none")
}

draw_circle(1,2,2)
```



, , .

Chapter 4

Methods

We describe our methods in this chapter.

Chapter 5

Applications

Some *significant* applications are demonstrated in this chapter.

5.1 Example one

5.2 Example two

Chapter 6

Final Words

We have finished a nice book.

Bibliography

(2020). : .