CS5800 Final Project

Lindsey Alexanian & Arushi Agarwal

CS5800 Final Project

Here is a visualization of mazes generated and solved in our project. The mazes themselves were generated using an implementation of Kruskal's algorithm in Python, and the shortest path from start to finish was found with BFS and DFS implementations in Python. Below are graphical visualizations of three mazes randomly generated.

Package installation

```r
install.packages("devtools")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```r
library(devtools)
```

```
## Loading required package: usethis
```

```r
install.packages("shiny")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```r
library(shiny)
```

DF

Here I generated a data frame with all possible points in the maze, just to use for labeling the graphs.

```r
# generate data frame for all points with x or y in (0:4), as this is a 5x5 maze
x <- 0:4
y <- 0:4
df <- data.frame(matrix(ncol=2, nrow=1))
for (x_val in x) {
  for (y_val in y) {
    df <- rbind(df, c(x_val, y_val))
  }
}

# remove empty row
df <- df[-1,]

# add row and column names
rownames(df) <- letters[1:nrow(df)]
names(df) <- c("x_value", "y_value")
```
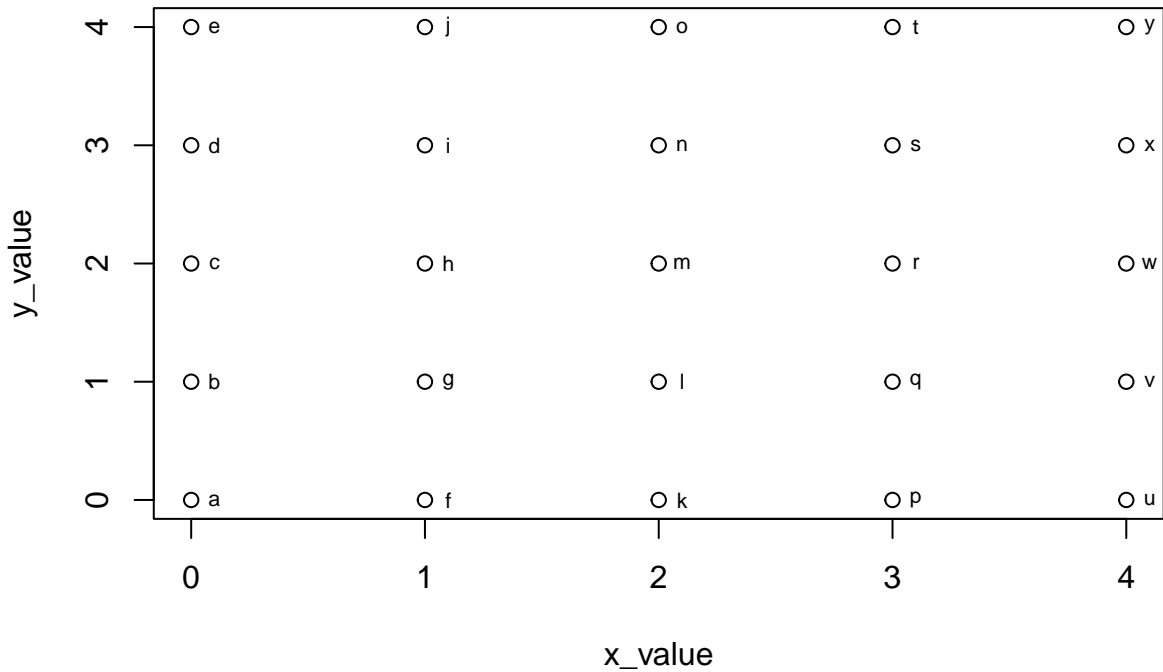
```
# test by making df
plot(df)
# add appropriate labels
text(df[,1] + 0.10, df[,2], labels=rownames(df), cex=0.7)
```


Maze 1

```
# individual maze points (generated from python code)
maze_points <- read.table("maze_edges.txt")

# maze edges (generated from python code)
edges <- read.table("maze_coordinate_pairs.txt")

# plot it
plot(0:6, 0:6, type = "n", bty="n", main="Maze 1", xlab="", ylab="") + points(maze_points, pch=19, col =
        col = "lightskyblue", lty = par("lty"), xpd = FALSE) + points(maze_points[1,], pch=19, col = "gr
```
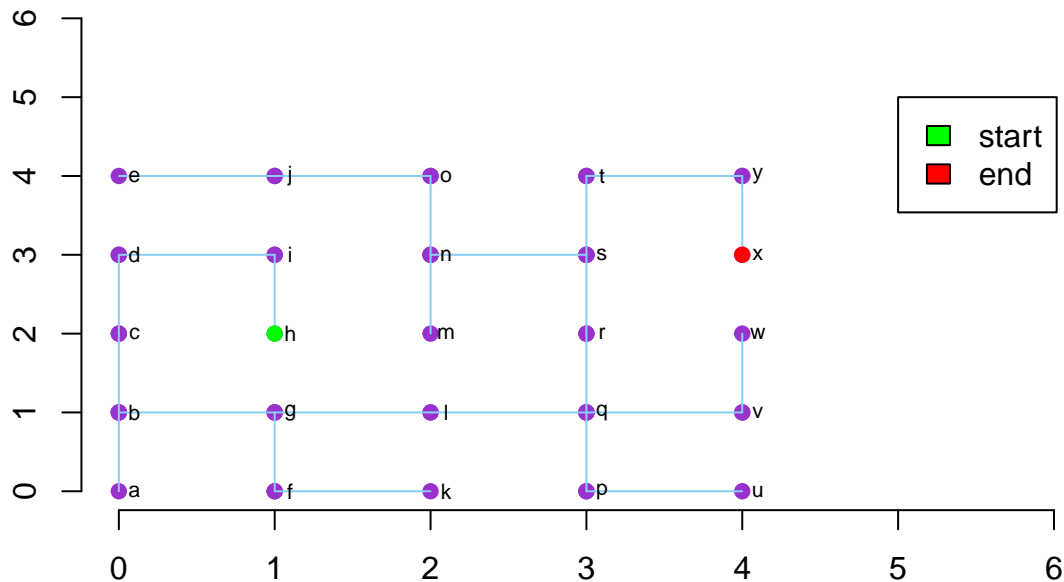
```
## integer(0)
```

```
# add labels to each point
text(df[,1] + 0.10, df[,2], labels=rownames(df), cex=0.7)

# add legend
legend(5, 5, legend=c("start", "end"),
        fill = c("green","red"))
```

Maze 1

Maze 2

```r
# get points from maze
maze_points2 <- read.table("maze_edges2.txt")
# get maze edges
edges2 <- read.table("maze_coordinate_pairs2.txt")

# plot it!
plot(0:6, 0:6, type = "n", bty="n", main="Maze 2", xlab="", ylab="") + points(maze_points2, pch=19, col
       col = "lightskyblue", lty = par("lty"), xpd = FALSE) + points(maze_points2[1,], pch=19, col = "g
```
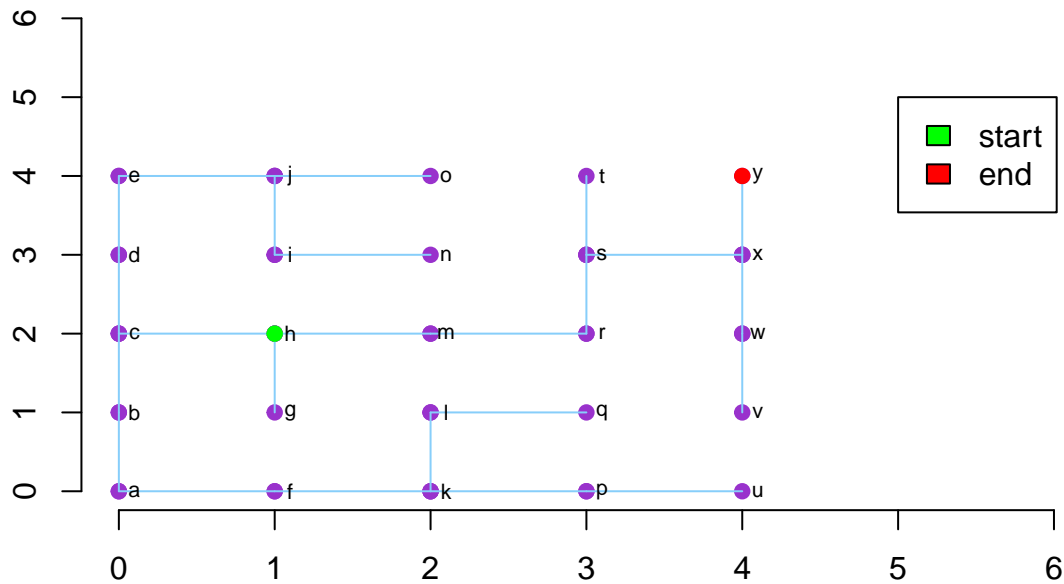
```
## integer(0)
```

```r
# add labels to points
text(df[,1] + 0.10, df[,2], labels=rownames(df), cex=0.7)

# add a legend
legend(5, 5, legend=c("start", "end"),
       fill = c("green","red"))
```

Maze 2

Maze 3

```r
# individual maze points (generated from python code)
maze_points3 <- read.table("maze_edges3.txt")

# maze edges (generated from python code)
edges3 <- read.table("maze_coordinate_pairs3.txt")

# plot it
plot(0:6, 0:6, type = "n", bty="n", main="Maze 3", xlab="", ylab="") + points(maze_points3, pch=19, col
        col = "lightskyblue", lty = par("lty"), xpd = FALSE) + points(maze_points3[1,], pch=19, col = "g
```

```
## integer(0)
```

```r
# add labels to each point
text(df[,1] + 0.10, df[,2], labels=rownames(df), cex=0.7)

# add legend
legend(5, 5, legend=c("start", "end"),
        fill = c("green","red"))
```

Maze 3