Creating CONSORT Diagrams in R

Jacky Choi, MPH

jmc4005@med.cornell.edu

2021-06-29

A Trip Down Memory Lane...

Overview

- CONSORT Statement and Diagram

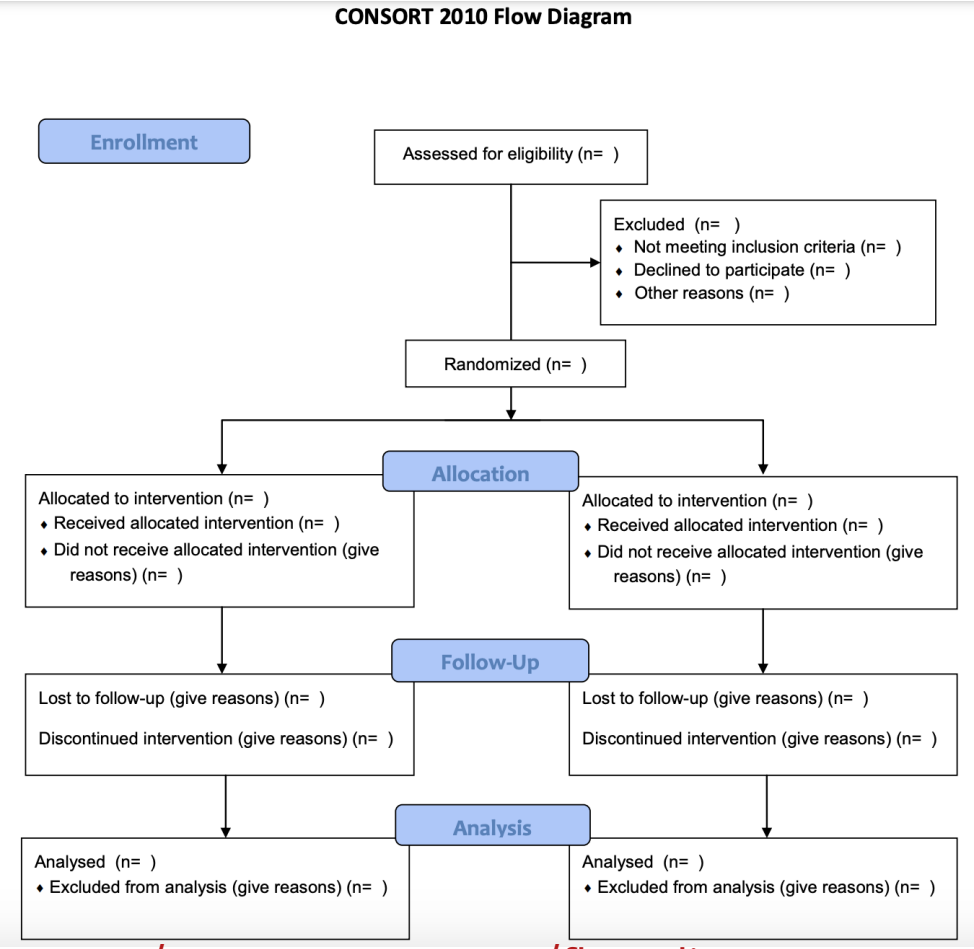
- DiagrammeR Package
  - GraphViz and DOT
  - Nodes, Edges, Graph Capabilities

- Building One, Yay!

About CONSORT Diagrams

- **CONSORT** (Consolidated Standards of Reporting Trials) **Statement**
- Guide to help researchers effectively report randomized trials to enhance transparency of study design and analysis
- For 2010, includes a 25-item checklist and a diagram showing participant flow
- Reporting participant flow helps assess the quality of the study design and reliability of the results

Example[1]



1 http://www.consort-statement.org/consort-statement/flow-diagram

DiagrammeR Package

- Powerful R package that allows you to create various diagrams and graphs
- Functionality includes GraphViz and DOT (graph description) language
- Pass various graph specifications to `grViz()` function using DOT language
- DOT is generally customizable and syntax is generally straightforward
- GraphViz substitution allows for mixing in R expressions

DiagrammeR Package

- Powerful R package that allows you to create various diagrams and graphs
- Functionality includes GraphViz and DOT (graph description) language
- Pass various graph specifications to `grViz()` function using DOT language
- DOT is generally customizable and syntax is generally straightforward
- GraphViz substitution allows for mixing in R expressions

Let's dive a bit deeper into the GraphViz specifications!

GraphViz Specifications

Graph statement

Define overall attributes of the visual you are creating

- `graph [nodesep = 0.8]`

Node statement

Define attributes for nodes, styled as boxes, circles, and other shapes

- `node [shape = box, width = 2] A; B; C; D`

Edge statement

Define attributes for the edges that connect nodes

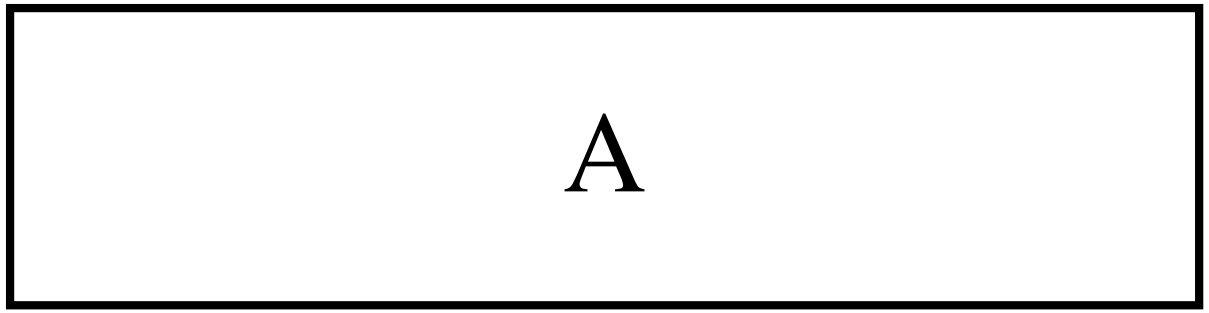
- `edge [arrowhead = diamonds] A -> B; B -> C; C -> D [label = 'Hello!']`

Basic Example

```
grViz("
digraph basic_example {

#Add nodes

#Add edges

}"
)
```

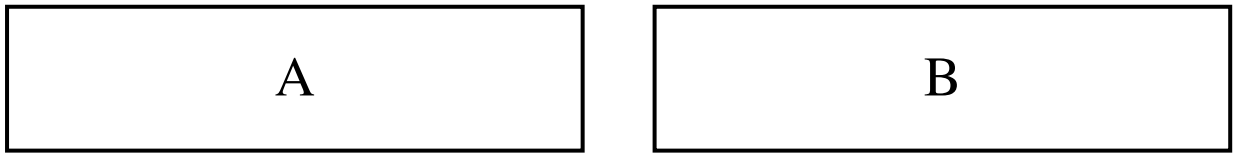
Basic Example

```
grViz("
digraph basic_example {

#Add nodes
node [shape = box, width = 2] #BREAK2
A #BREAK2

#Add edges

}"
)
```



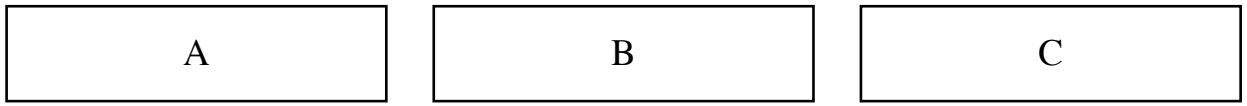
Basic Example

```
grViz("
digraph basic_example {

#Add nodes
node [shape = box, width = 2] #BREAK2
A #BREAK2
B #BREAK3

#Add edges

}"
)
```



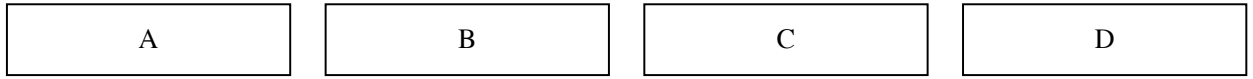
Basic Example

```
grViz("
digraph basic_example {

#Add nodes
node [shape = box, width = 2] #BREAK2
A #BREAK2
B #BREAK3
C #BREAK4

#Add edges

}"
)
```



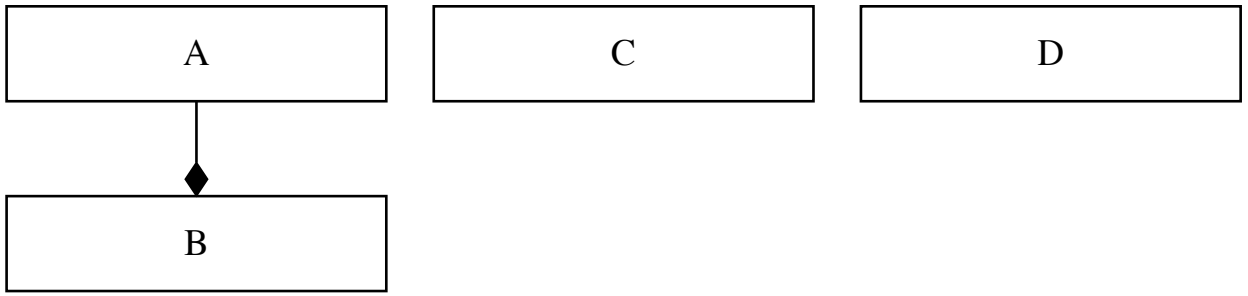
Basic Example

```
grViz("
digraph basic_example {

#Add nodes
node [shape = box, width = 2] #BREAK2
A #BREAK2
B #BREAK3
C #BREAK4
D #BREAK5

#Add edges

}"
)
```



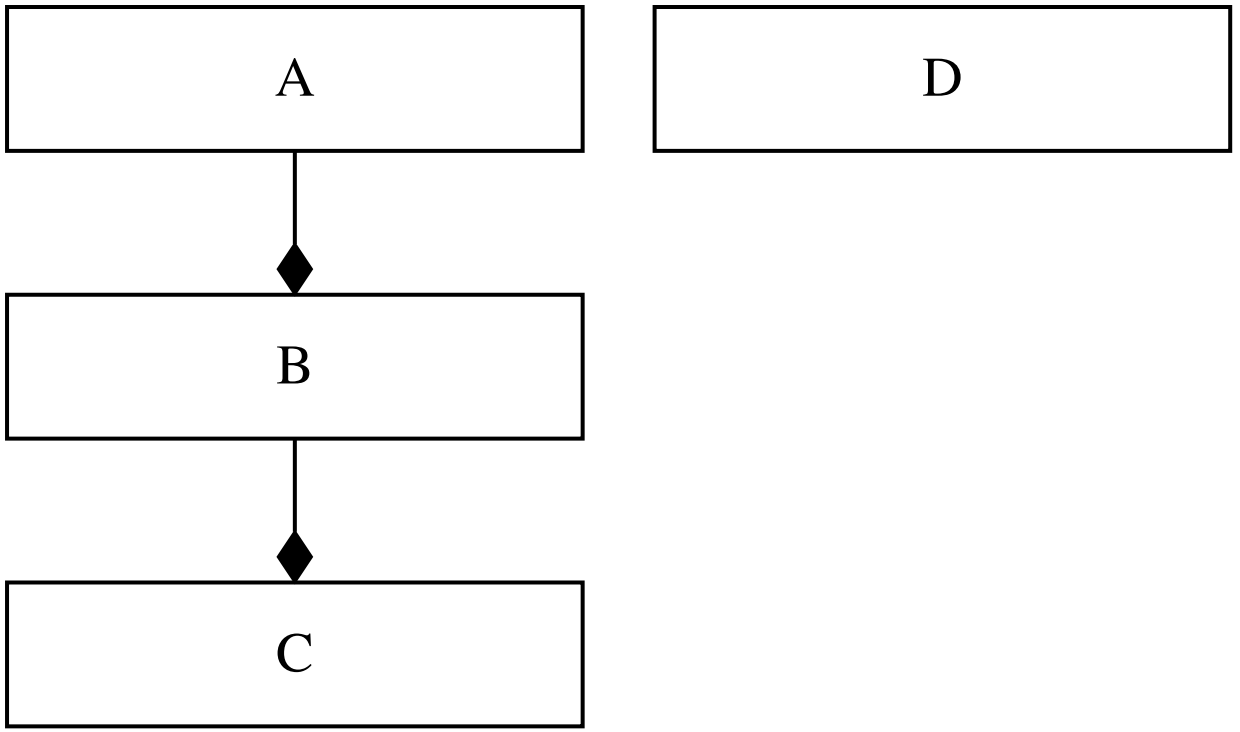
Basic Example

```
grViz("
digraph basic_example {

#Add nodes
node [shape = box, width = 2] #BREAK2
A #BREAK2
B #BREAK3
C #BREAK4
D #BREAK5

#Add edges
edge [arrowhead = diamonds] #BREAK6
A -> B #BREAK6

}"
)
```



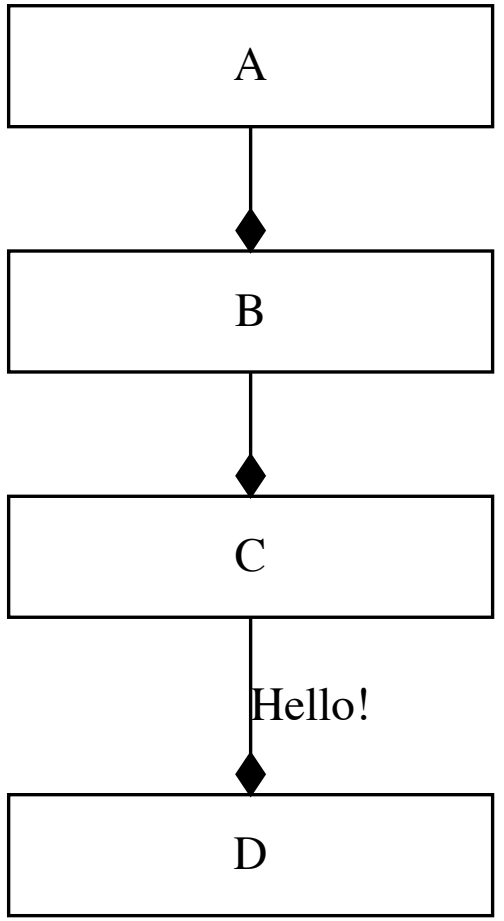
Basic Example

```
grViz("
digraph basic_example {

#Add nodes
node [shape = box, width = 2] #BREAK2
A #BREAK2
B #BREAK3
C #BREAK4
D #BREAK5

#Add edges
edge [arrowhead = diamonds] #BREAK6
A -> B #BREAK6
B -> C #BREAK7

}"
)
```



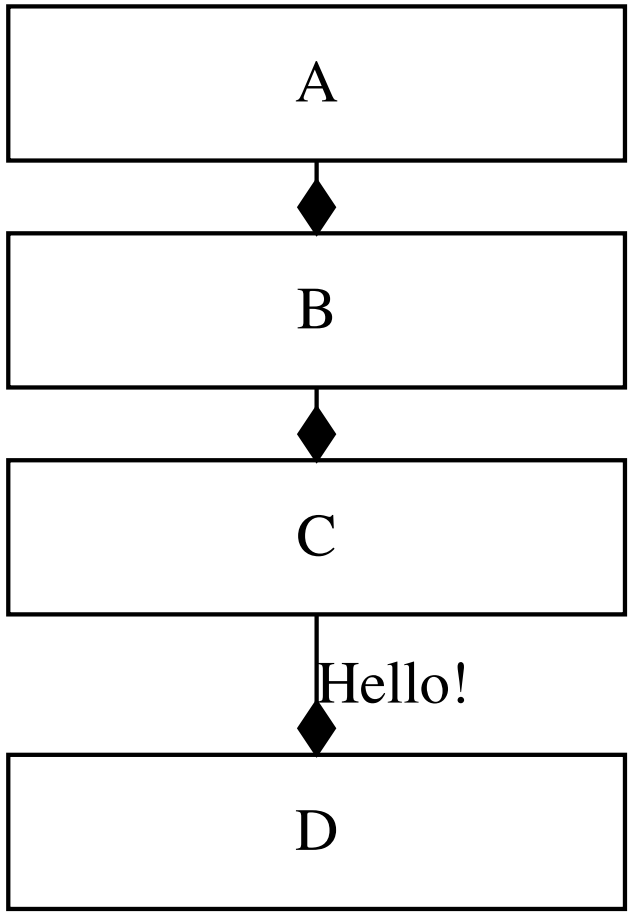
Basic Example

```
grViz("
digraph basic_example {

#Add nodes
node [shape = box, width = 2] #BREAK2
A #BREAK2
B #BREAK3
C #BREAK4
D #BREAK5

#Add edges
edge [arrowhead = diamonds] #BREAK6
A -> B #BREAK6
B -> C #BREAK7
C -> D [label = 'Hello!'] #BREAK8

}"
)
```



Basic Example

```
grViz("
digraph basic_example {

#Add nodes
node [shape = box, width = 2] #BREAK2
A #BREAK2
B #BREAK3
C #BREAK4
D #BREAK5

#Add edges
edge [arrowhead = diamonds] #BREAK6
A -> B #BREAK6
B -> C #BREAK7
C -> D [label = 'Hello!'] #BREAK8

#Add graph statement #BREAK9
graph [ranksep = 0.1] #BREAK9
}"
)
```



Example Trial Data

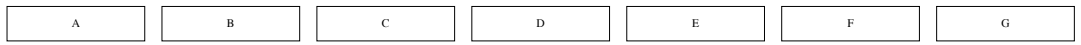
- 2-arm randomized controlled trial comparing care from a home nurse plus standard prenatal care versus standard prenatal care for pregnant women with high glucose levels
- Primary outcome was composite measure of perinatal death and neonatal complications
- 843 pregnant women were screened and assessed for eligibility at doctors' offices
  - 26 did not meet inclusion criteria; 36 declined to participate; 11 could not be reached
- 770 were eligible and randomized
- 391 were assigned to treatment group; 379 were assigned to control group
  - Treatment: 21/391 were lost to follow-up and 8/391 had no delivery data
  - Control: 17/379 were lost to follow-up and 10/379 had no delivery data

Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes

#Add edges

#Add graph statement
}"
)
```

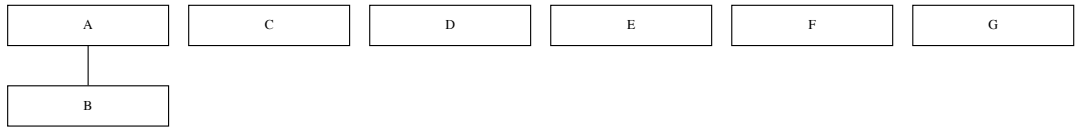
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges

#Add graph statement
}"
)
```



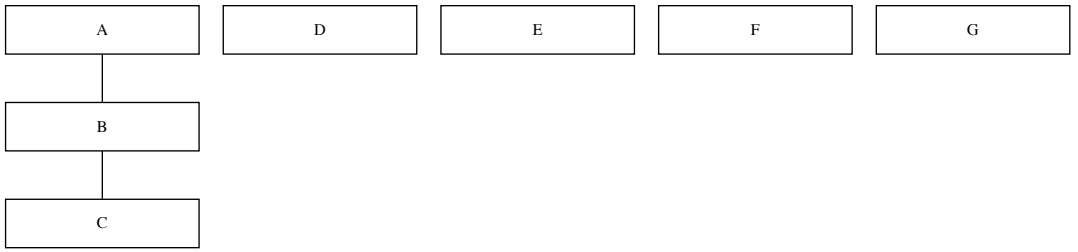
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3

#Add graph statement
}"
)
```



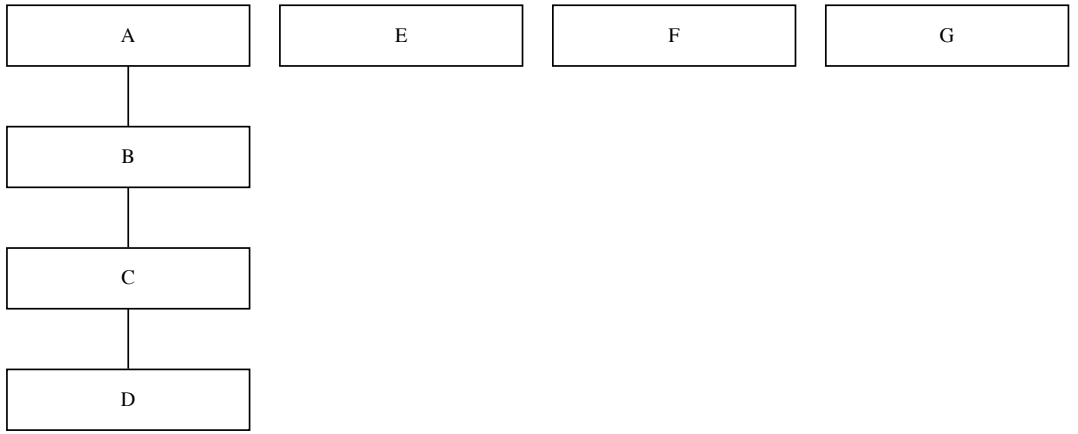
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4

#Add graph statement
}"
)
```



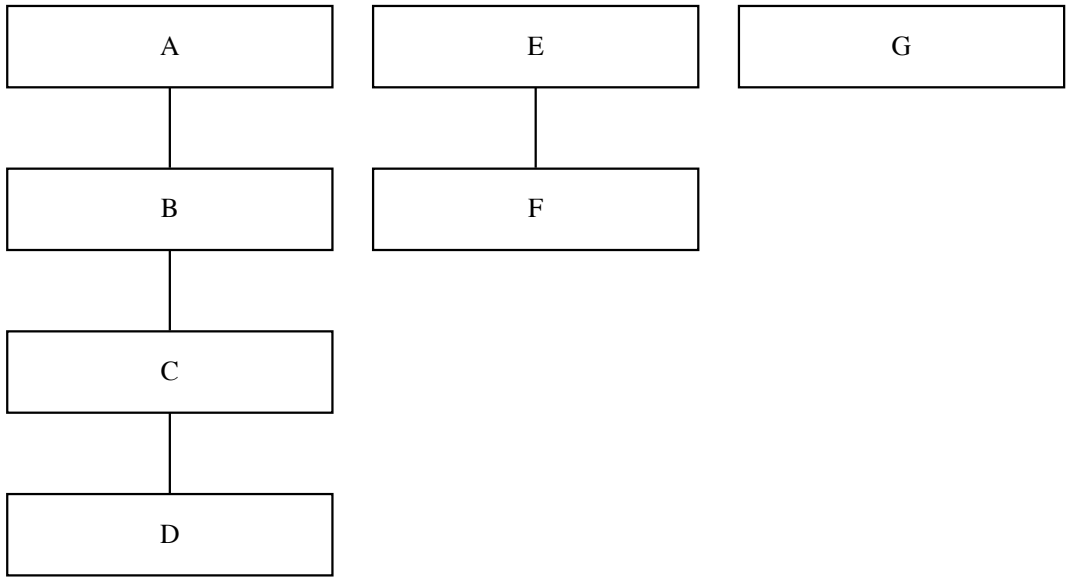
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4
C -> D #BREAK5

#Add graph statement
}"
)
```



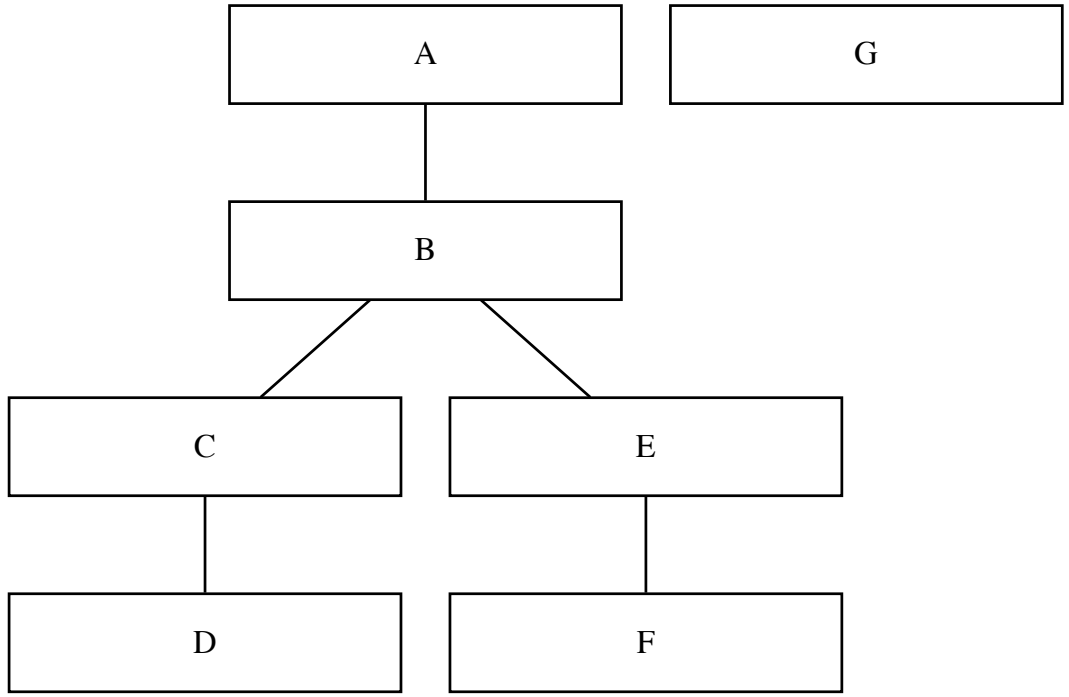
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4
C -> D #BREAK5
E -> F #BREAK6

#Add graph statement
}"
)
```



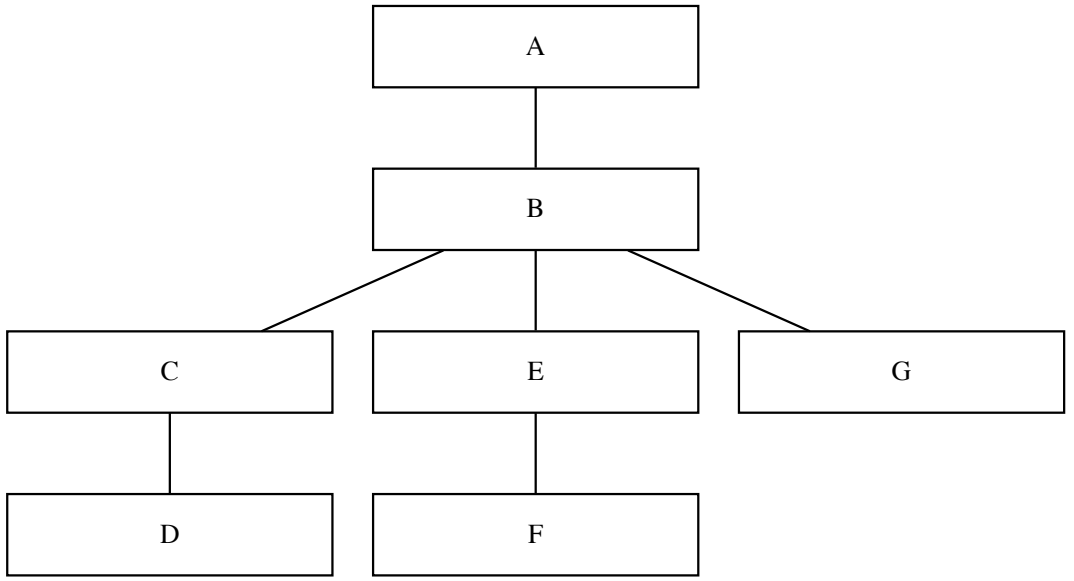
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4
C -> D #BREAK5
E -> F #BREAK6
B -> E #BREAK7

#Add graph statement
}"
)
```



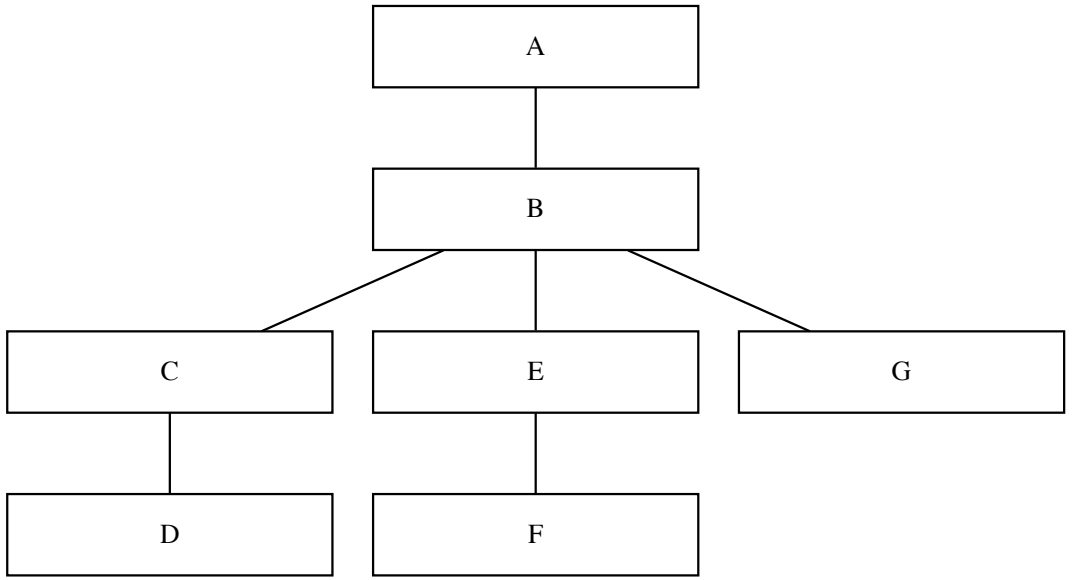
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4
C -> D #BREAK5
E -> F #BREAK6
B -> E #BREAK7
B -> G #BREAK8

#Add graph statement
}"
)
```



Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4
C -> D #BREAK5
E -> F #BREAK6
B -> E #BREAK7
B -> G #BREAK8

#Add graph statement
graph [ranksep = 0.5] #BREAK9
}"
)
```

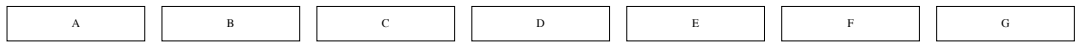


Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes

#Add edges

#Add graph statement
}"
)
```

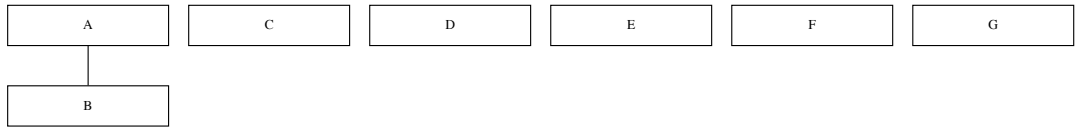
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges

#Add graph statement
}"
)
```



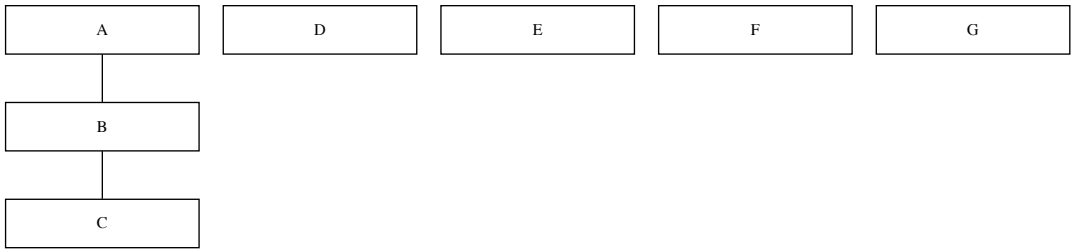
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3

#Add graph statement
}"
)
```



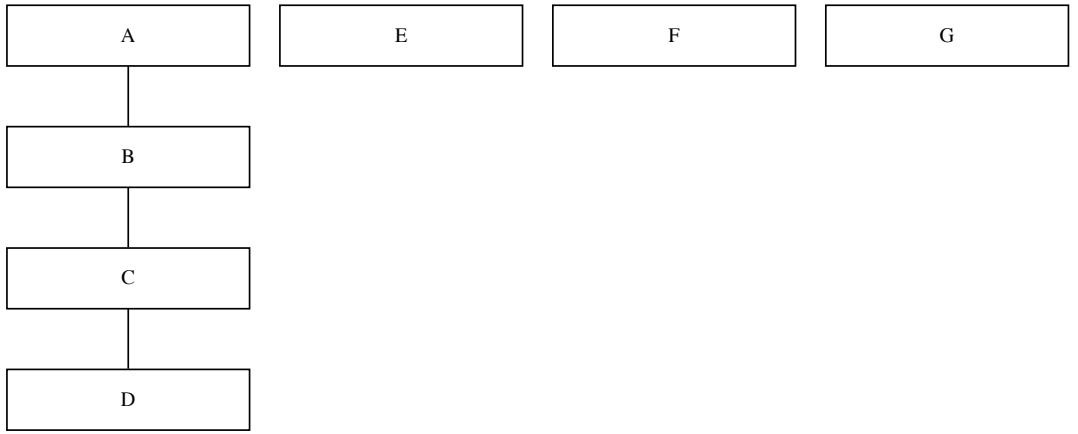
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4

#Add graph statement
}"
)
```



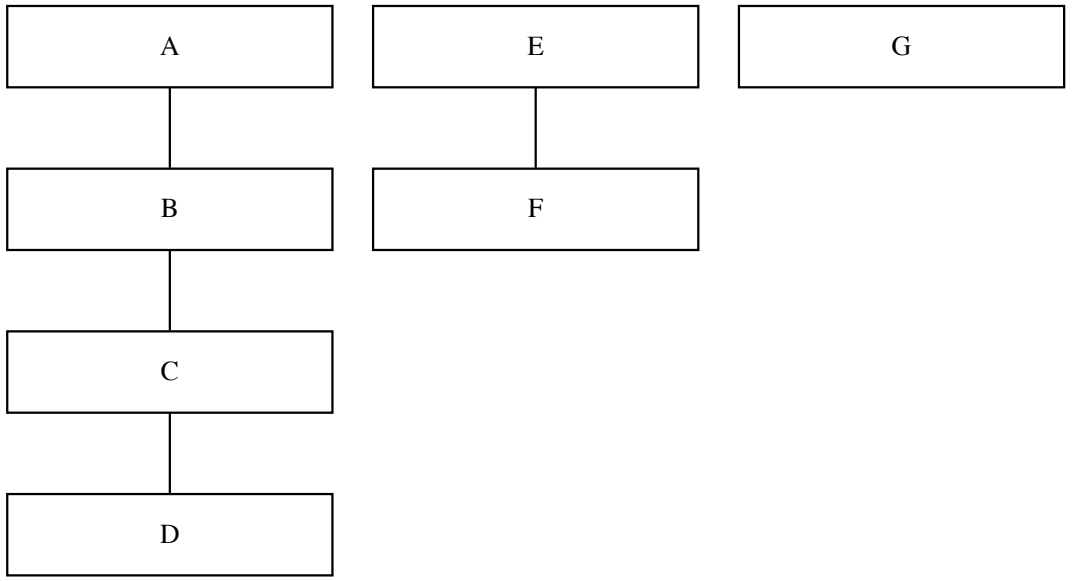
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4
C -> D #BREAK5

#Add graph statement
}"
)
```



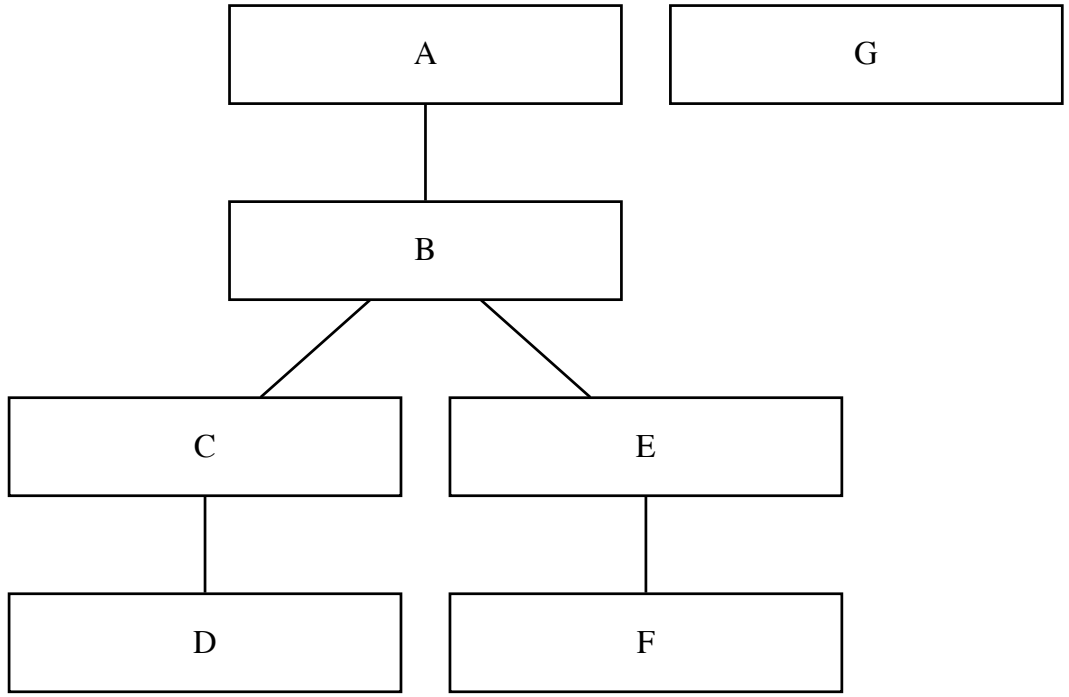
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4
C -> D #BREAK5
E -> F #BREAK6

#Add graph statement
}"
)
```



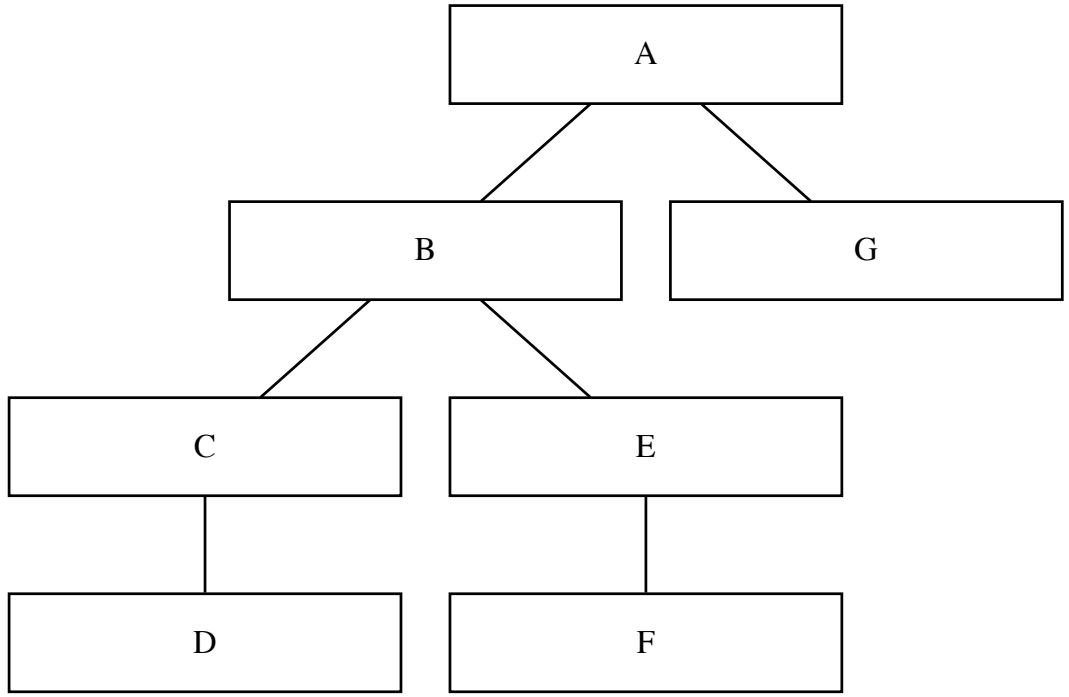
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4
C -> D #BREAK5
E -> F #BREAK6
B -> E #BREAK7

#Add graph statement
}"
)
```



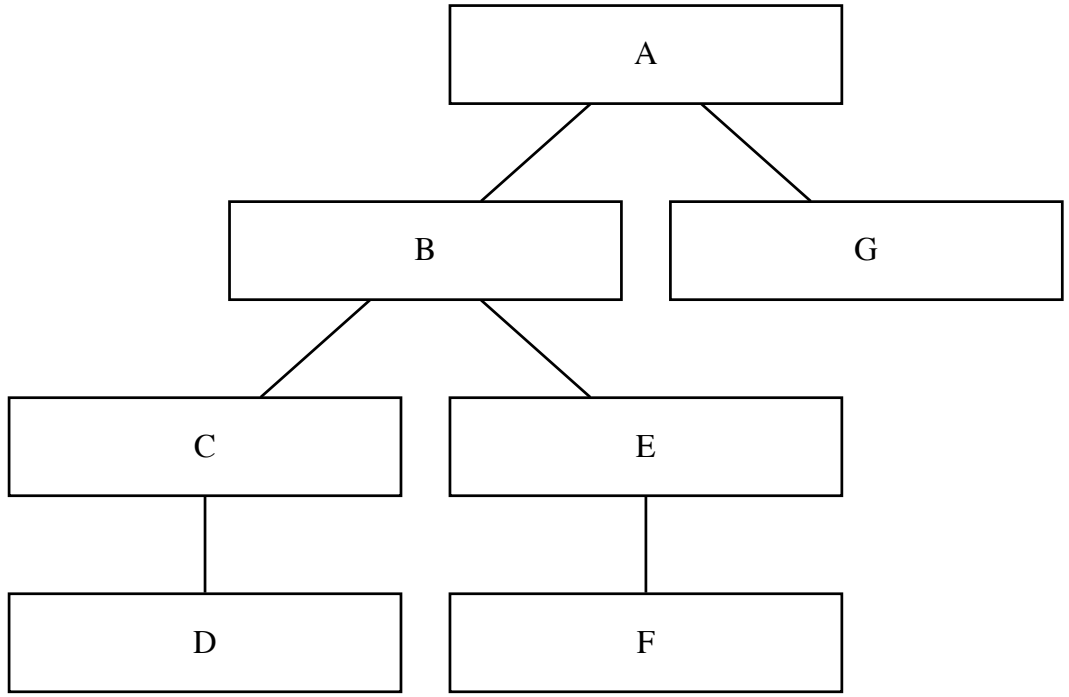
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4
C -> D #BREAK5
E -> F #BREAK6
B -> E #BREAK7
A -> G #BREAK8

#Add graph statement
}"
)
```



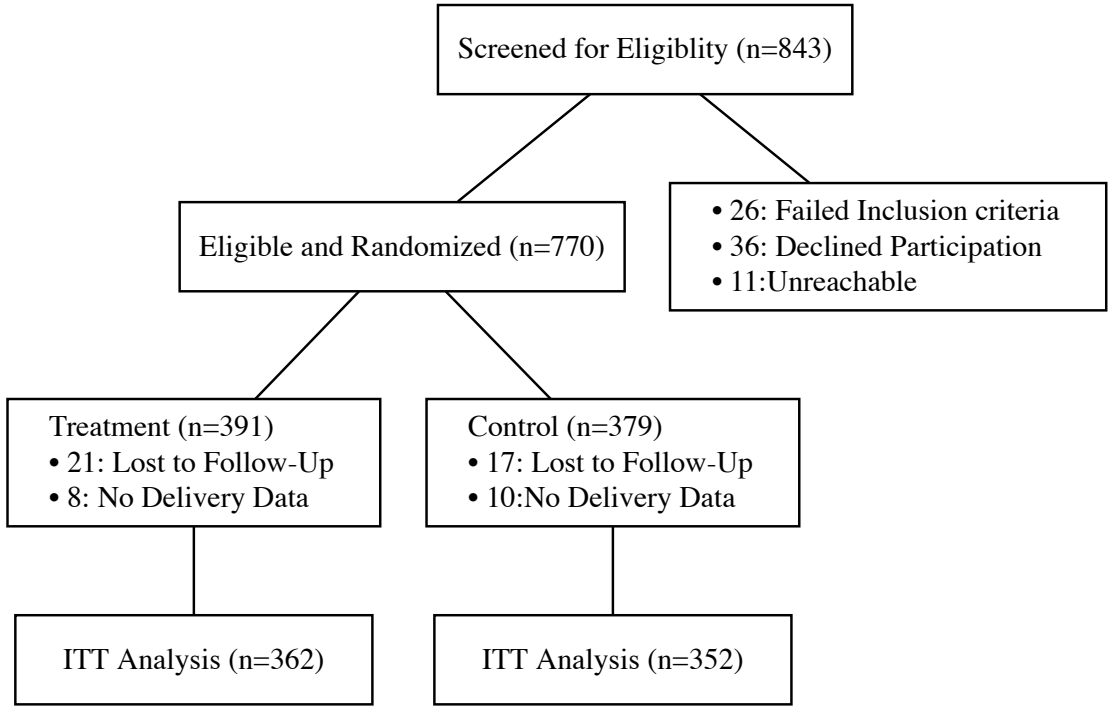
Attempt 1 - Build Skeleton

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = black, wid
A; B; C; D; E; F; G #BREAK2

#Add edges
edge [arrowhead = none] #BREAK3
A -> B #BREAK3
B -> C #BREAK4
C -> D #BREAK5
E -> F #BREAK6
B -> E #BREAK7
A -> G #BREAK8

#Add graph statement
graph [ranksep = 0.5] #BREAK9
}"
)
```



Attempt 1 - Add in Labels

```
grViz("
digraph attempt_1 {
#Add nodes
node [shape = box, fontsize = 12, color = blacl
A [label = 'Screened for Eligiblity (n=843)']
B [label = 'Eligible and Randomized (n=770)']
C [label = <
    Treatment (n=391)<br ALIGN = 'LEFT'/>
    &#8226; 21: Lost to Follow-Up<br ALIGN = '
    &#8226; 8: No Delivery Data<br ALIGN = 'LE
    >]
D [label = 'ITT Analysis (n=362)']
E [label = <
    Control (n=379)<br ALIGN = 'LEFT'/>
    &#8226; 17: Lost to Follow-Up<br ALIGN = '
    &#8226; 10:No Delivery Data<br ALIGN = 'LE
    >]
F [label = 'ITT Analysis (n=352)']
G [label = <
    &#8226; 26: Failed Inclusion criteria<br A
    &#8226; 36: Declined Participation<br ALIGI
```



About That...

- We were nearly able to obtain our desired result
- GraphViz generally does a great job connecting edges to nodes, but needs help from time to time
- We want edges to be completely vertical or horizontal
- We don't want the diagram to be angled or tilted from the addition of any side nodes
- Try to visualize as a grid

About That...

- We were nearly able to obtain our desired result
- GraphViz generally does a great job connecting edges to nodes, but needs help from time to time
- We want edges to be completely vertical or horizontal
- We don't want the diagram to be angled or tilted from the addition of any side nodes
- Try to visualize as a grid

Let's try again!

Questions?

Attempt 2 - Build Skeleton

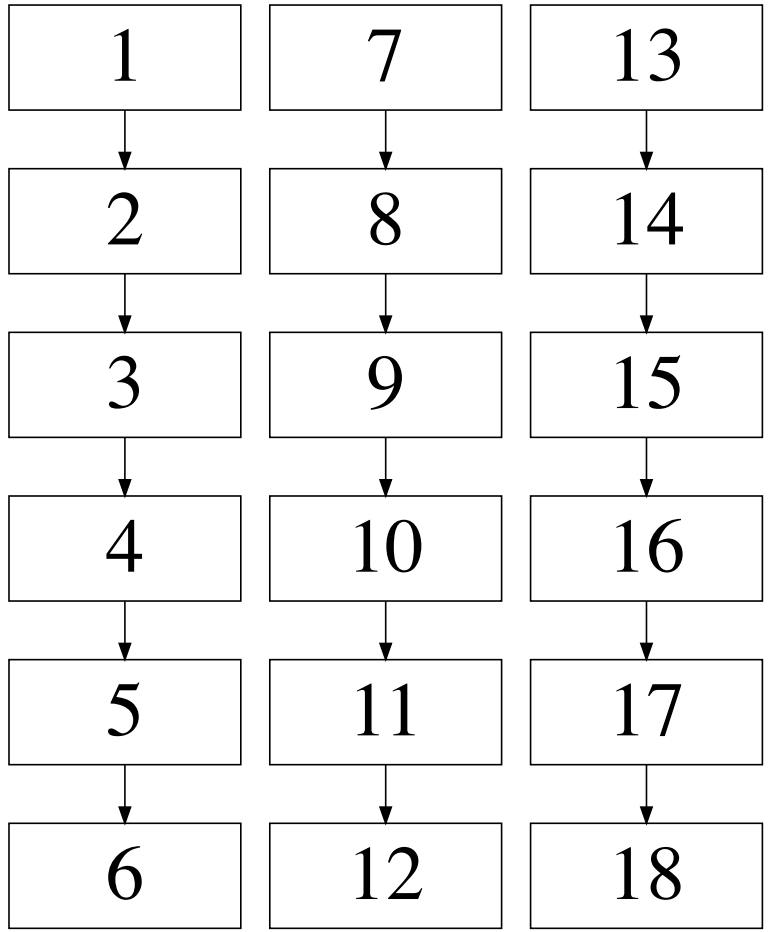
```
grViz("
digraph {
#Add nodes

#Add edges

#Add additional arrowheads


#Use rank to keep these three boxes on the same rank

#Use nodesep to keep the nodes a bit closer together

}"
)
```

Attempt 2 - Build Skeleton

```
grViz("
digraph {
#Add nodes
node [style = 'solid', shape = 'box', width = 2, fon
1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 1

#Add edges

#Add additional arrowheads


#Use rank to keep these three boxes on the same rank

#Use nodesep to keep the nodes a bit closer together

}"
)
```

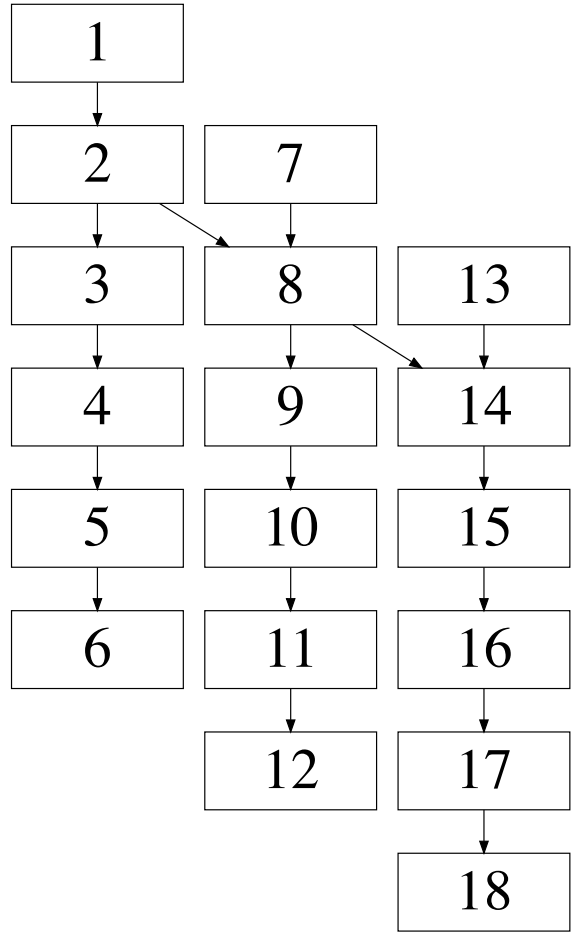
Attempt 2 - Build Skeleton

```
grViz("
digraph {
#Add nodes
node [style = 'solid', shape = 'box', width = 2, fon
1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 1

#Add edges
edge [arrowhead = 'normal', color = 'black'] #BREAK3
1 -> 2; 2 -> 3; 3 -> 4; 4 -> 5; 5 -> 6; 7 -> 8; 8 ->

#Add additional arrowheads


#Use rank to keep these three boxes on the same rank

#Use nodesep to keep the nodes a bit closer together

}"
)
```



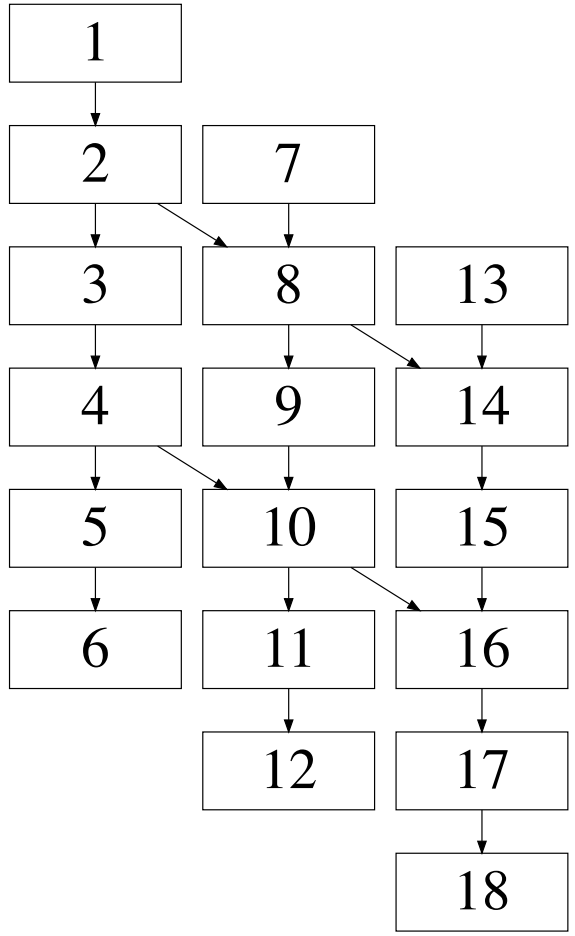
Attempt 2 - Build Skeleton

```
grViz("
digraph {
#Add nodes
node [style = 'solid', shape = 'box', width = 2, fon
1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 1

#Add edges
edge [arrowhead = 'normal', color = 'black'] #BREAK3
1 -> 2; 2 -> 3; 3 -> 4; 4 -> 5; 5 -> 6; 7 -> 8; 8 ->

#Add additional arrowheads
edge [arrowhead = 'normal', color = 'black'] #BREAK4
2 -> 8; 8 -> 14 #BREAK4


#Use rank to keep these three boxes on the same rank

#Use nodesep to keep the nodes a bit closer together

}"
)
```



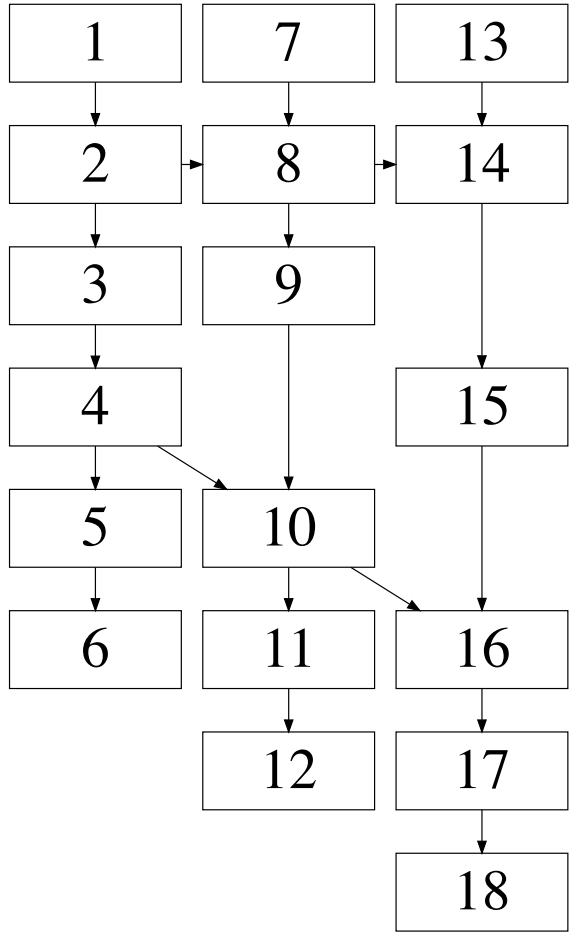
Attempt 2 - Build Skeleton

```
grViz("
digraph {
#Add nodes
node [style = 'solid', shape = 'box', width = 2, fon
1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 1

#Add edges
edge [arrowhead = 'normal', color = 'black'] #BREAK3
1 -> 2; 2 -> 3; 3 -> 4; 4 -> 5; 5 -> 6; 7 -> 8; 8 ->

#Add additional arrowheads
edge [arrowhead = 'normal', color = 'black'] #BREAK4
2 -> 8; 8 -> 14 #BREAK4

edge [arrowhead = 'normal', color = 'black'] #BREAK5
4 -> 10; 10 -> 16 #BREAK5

#Use rank to keep these three boxes on the same rank

#Use nodesep to keep the nodes a bit closer together

}"
```



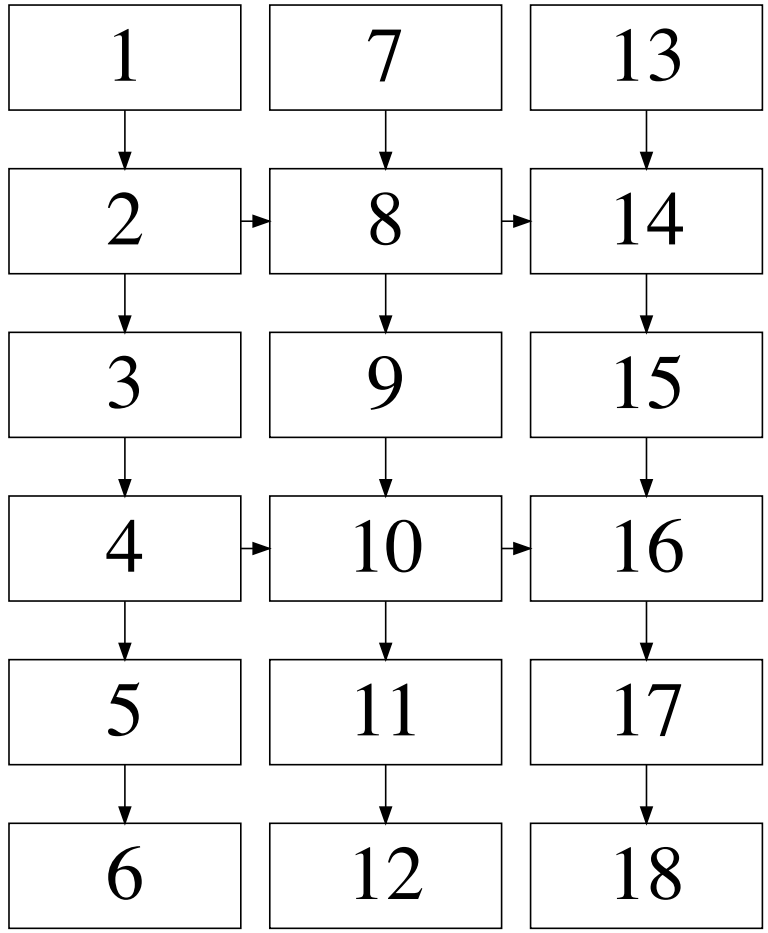
Attempt 2 - Build Skeleton

```
grViz("
digraph {
#Add nodes
node [style = 'solid', shape = 'box', width = 2, fon
1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 1

#Add edges
edge [arrowhead = 'normal', color = 'black'] #BREAK3
1 -> 2; 2 -> 3; 3 -> 4; 4 -> 5; 5 -> 6; 7 -> 8; 8 ->

#Add additional arrowheads
edge [arrowhead = 'normal', color = 'black'] #BREAK4
2 -> 8; 8 -> 14 #BREAK4

edge [arrowhead = 'normal', color = 'black'] #BREAK5
4 -> 10; 10 -> 16 #BREAK5

#Use rank to keep these three boxes on the same rank
{rank=same; '2';'8';'14'} #BREAK6

#Use nodesep to keep the nodes a bit closer together
```



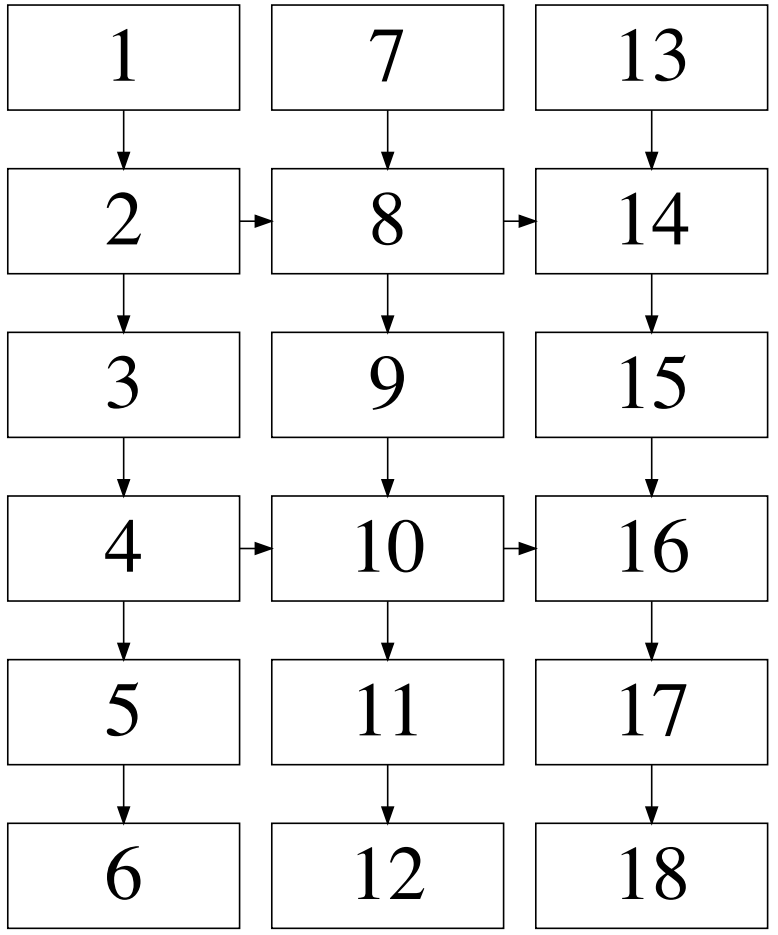
Attempt 2 - Build Skeleton

```
grViz("
digraph {
#Add nodes
node [style = 'solid', shape = 'box', width = 2, fon
1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 1

#Add edges
edge [arrowhead = 'normal', color = 'black'] #BREAK3
1 -> 2; 2 -> 3; 3 -> 4; 4 -> 5; 5 -> 6; 7 -> 8; 8 ->

#Add additional arrowheads
edge [arrowhead = 'normal', color = 'black'] #BREAK4
2 -> 8; 8 -> 14 #BREAK4

edge [arrowhead = 'normal', color = 'black'] #BREAK5
4 -> 10; 10 -> 16 #BREAK5

#Use rank to keep these three boxes on the same rank
{rank=same; '2';'8';'14'} #BREAK6
{rank=same; '4';'10';'16'} #BREAK7

#Use nodesep to keep the nodes a bit closer together
```



Attempt 2 - Build Skeleton

```
grViz("
digraph {
#Add nodes
node [style = 'solid', shape = 'box', width = 2, fon
1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 1

#Add edges
edge [arrowhead = 'normal', color = 'black'] #BREAK3
1 -> 2; 2 -> 3; 3 -> 4; 4 -> 5; 5 -> 6; 7 -> 8; 8 ->

#Add additional arrowheads
edge [arrowhead = 'normal', color = 'black'] #BREAK4
2 -> 8; 8 -> 14 #BREAK4

edge [arrowhead = 'normal', color = 'black'] #BREAK5
4 -> 10; 10 -> 16 #BREAK5

#Use rank to keep these three boxes on the same rank
{rank=same; '2';'8';'14'} #BREAK6
{rank=same; '4';'10';'16'} #BREAK7

#Use nodesep to keep the nodes a bit closer together
```

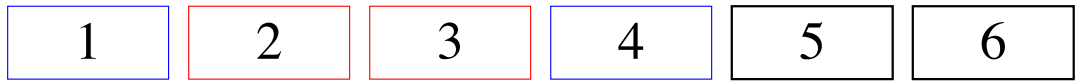


Attempt 2 - Define Grid

```
grViz("
digraph {

#First column

#Second column

#Third column

#Add edges

#Add additional arrowheads


#Use rank to keep these three boxes on the same rank

#Use nodesep to keep the nodes a bit closer together

}"
)
```

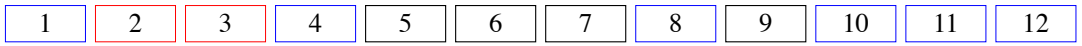
Attempt 2 - Define Grid

```
grViz("
digraph {

#First column
1 [style = 'solid', shape = 'box', width = 2, fontsi
2 [style = 'solid', shape = 'box', width = 2, fontsi
3 [style = 'solid', shape = 'box', width = 2, fontsi
4 [style = 'solid', shape = 'box', width = 2, fontsi
5 [style = 'solid', shape = 'box', width = 2, fontsi
6 [style = 'solid', shape = 'box', width = 2, fontsi

#Second column

#Third column

#Add edges

#Add additional arrowheads


#Use rank to keep these three boxes on the same rank

#Use nodesep to keep the nodes a bit closer together
```



Attempt 2 - Define Grid

```
grViz("
digraph {

#First column
1 [style = 'solid', shape = 'box', width = 2, fontsi
2 [style = 'solid', shape = 'box', width = 2, fontsi
3 [style = 'solid', shape = 'box', width = 2, fontsi
4 [style = 'solid', shape = 'box', width = 2, fontsi
5 [style = 'solid', shape = 'box', width = 2, fontsi
6 [style = 'solid', shape = 'box', width = 2, fontsi

#Second column
7 [style = 'solid', shape = 'box', width = 2, fontsi
8 [style = 'solid', shape = 'box', width = 2, fontsi
9 [style = 'solid', shape = 'box', width = 2, fontsi
10 [style = 'solid', shape = 'box', width = 2, fonts
11 [style = 'solid', shape = 'box', width = 2, fonts
12 [style = 'solid', shape = 'box', width = 2, fonts

#Third column

#Add edges
```

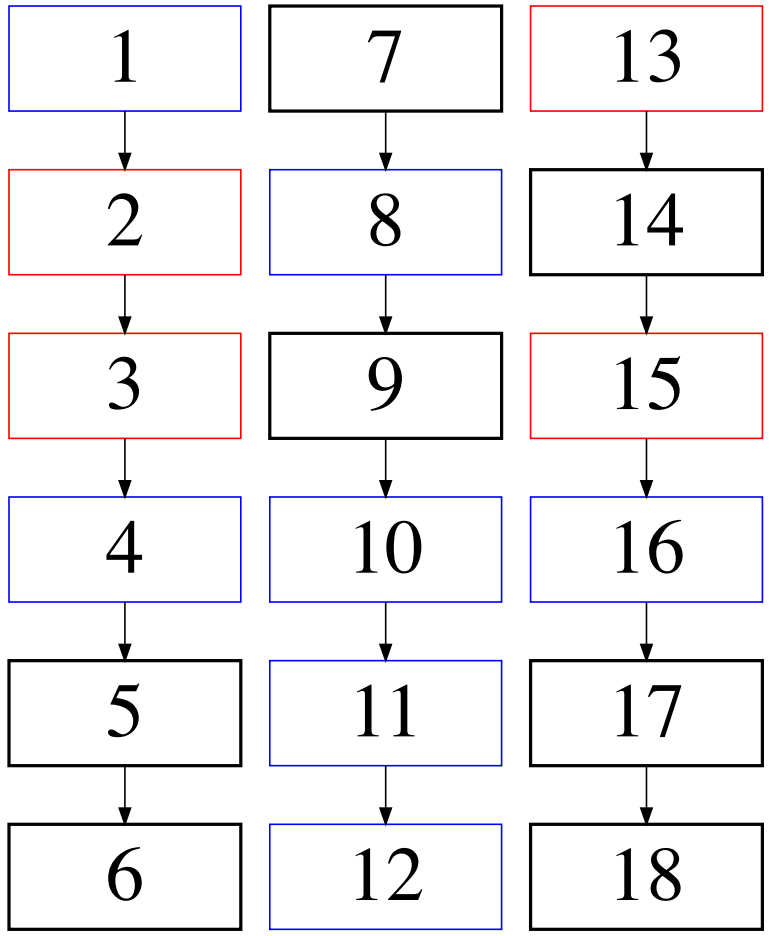


Attempt 2 - Define Grid

```
grViz("
digraph {

#First column
1 [style = 'solid', shape = 'box', width = 2, fontsi
2 [style = 'solid', shape = 'box', width = 2, fontsi
3 [style = 'solid', shape = 'box', width = 2, fontsi
4 [style = 'solid', shape = 'box', width = 2, fontsi
5 [style = 'solid', shape = 'box', width = 2, fontsi
6 [style = 'solid', shape = 'box', width = 2, fontsi

#Second column
7 [style = 'solid', shape = 'box', width = 2, fontsi
8 [style = 'solid', shape = 'box', width = 2, fontsi
9 [style = 'solid', shape = 'box', width = 2, fontsi
10 [style = 'solid', shape = 'box', width = 2, fonts
11 [style = 'solid', shape = 'box', width = 2, fonts
12 [style = 'solid', shape = 'box', width = 2, fonts

#Third column
13 [style = 'solid', shape = 'box', width = 2, fonts
14 [style = 'solid', shape = 'box', width = 2, fonts
```

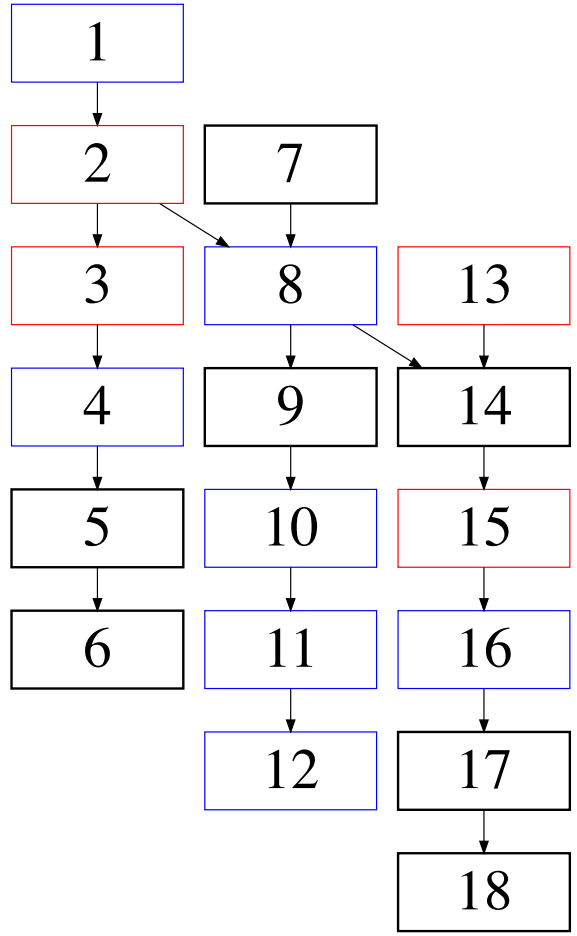
Attempt 2 - Define Grid

```
grViz("
digraph {

#First column
1 [style = 'solid', shape = 'box', width = 2, fontsi
2 [style = 'solid', shape = 'box', width = 2, fontsi
3 [style = 'solid', shape = 'box', width = 2, fontsi
4 [style = 'solid', shape = 'box', width = 2, fontsi
5 [style = 'solid', shape = 'box', width = 2, fontsi
6 [style = 'solid', shape = 'box', width = 2, fontsi

#Second column
7 [style = 'solid', shape = 'box', width = 2, fontsi
8 [style = 'solid', shape = 'box', width = 2, fontsi
9 [style = 'solid', shape = 'box', width = 2, fontsi
10 [style = 'solid', shape = 'box', width = 2, fonts
11 [style = 'solid', shape = 'box', width = 2, fonts
12 [style = 'solid', shape = 'box', width = 2, fonts

#Third column
13 [style = 'solid', shape = 'box', width = 2, fonts
14 [style = 'solid', shape = 'box', width = 2, fonts
```



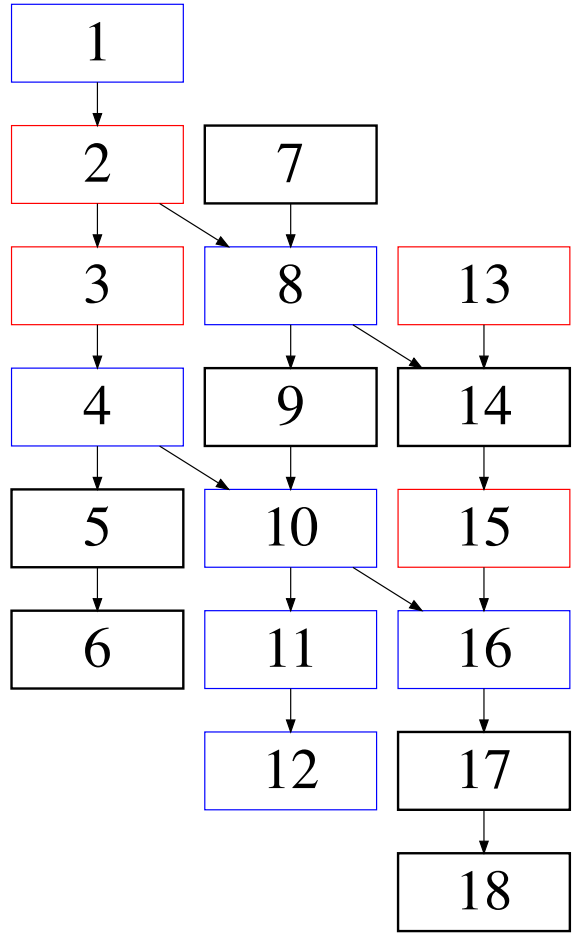
Attempt 2 - Define Grid

```
grViz("
digraph {

#First column
1 [style = 'solid', shape = 'box', width = 2, fontsi
2 [style = 'solid', shape = 'box', width = 2, fontsi
3 [style = 'solid', shape = 'box', width = 2, fontsi
4 [style = 'solid', shape = 'box', width = 2, fontsi
5 [style = 'solid', shape = 'box', width = 2, fontsi
6 [style = 'solid', shape = 'box', width = 2, fontsi

#Second column
7 [style = 'solid', shape = 'box', width = 2, fontsi
8 [style = 'solid', shape = 'box', width = 2, fontsi
9 [style = 'solid', shape = 'box', width = 2, fontsi
10 [style = 'solid', shape = 'box', width = 2, fonts
11 [style = 'solid', shape = 'box', width = 2, fonts
12 [style = 'solid', shape = 'box', width = 2, fonts

#Third column
13 [style = 'solid', shape = 'box', width = 2, fonts
14 [style = 'solid', shape = 'box', width = 2, fonts
```



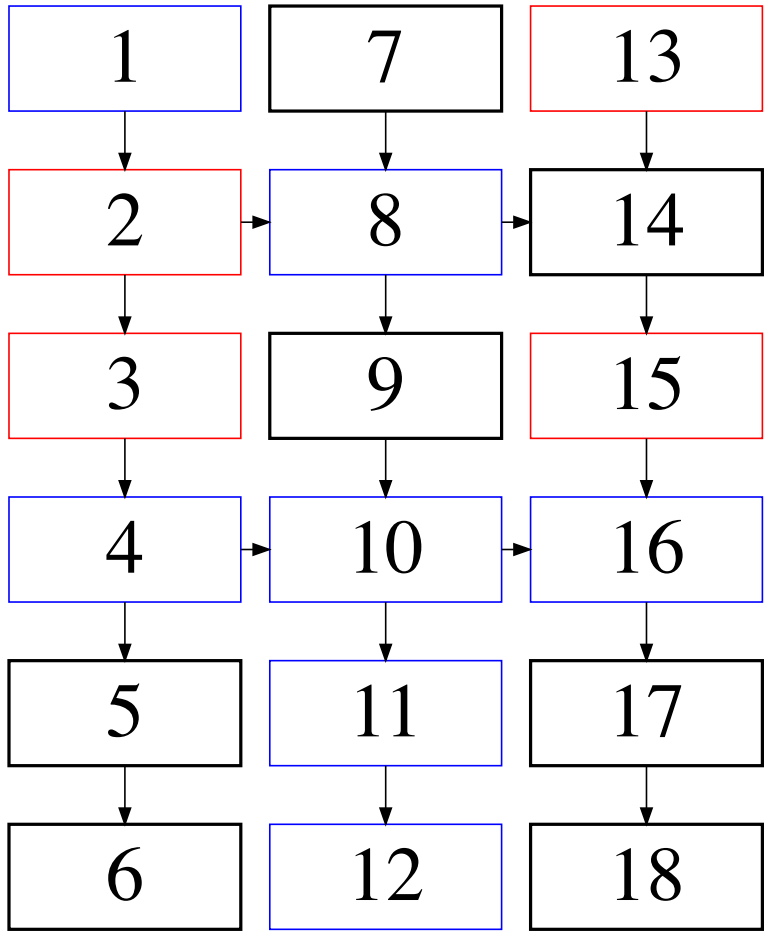
Attempt 2 - Define Grid

```
grViz("
digraph {

#First column
1 [style = 'solid', shape = 'box', width = 2, fontsi
2 [style = 'solid', shape = 'box', width = 2, fontsi
3 [style = 'solid', shape = 'box', width = 2, fontsi
4 [style = 'solid', shape = 'box', width = 2, fontsi
5 [style = 'solid', shape = 'box', width = 2, fontsi
6 [style = 'solid', shape = 'box', width = 2, fontsi

#Second column
7 [style = 'solid', shape = 'box', width = 2, fontsi
8 [style = 'solid', shape = 'box', width = 2, fontsi
9 [style = 'solid', shape = 'box', width = 2, fontsi
10 [style = 'solid', shape = 'box', width = 2, fonts
11 [style = 'solid', shape = 'box', width = 2, fonts
12 [style = 'solid', shape = 'box', width = 2, fonts

#Third column
13 [style = 'solid', shape = 'box', width = 2, fonts
14 [style = 'solid', shape = 'box', width = 2, fonts
```



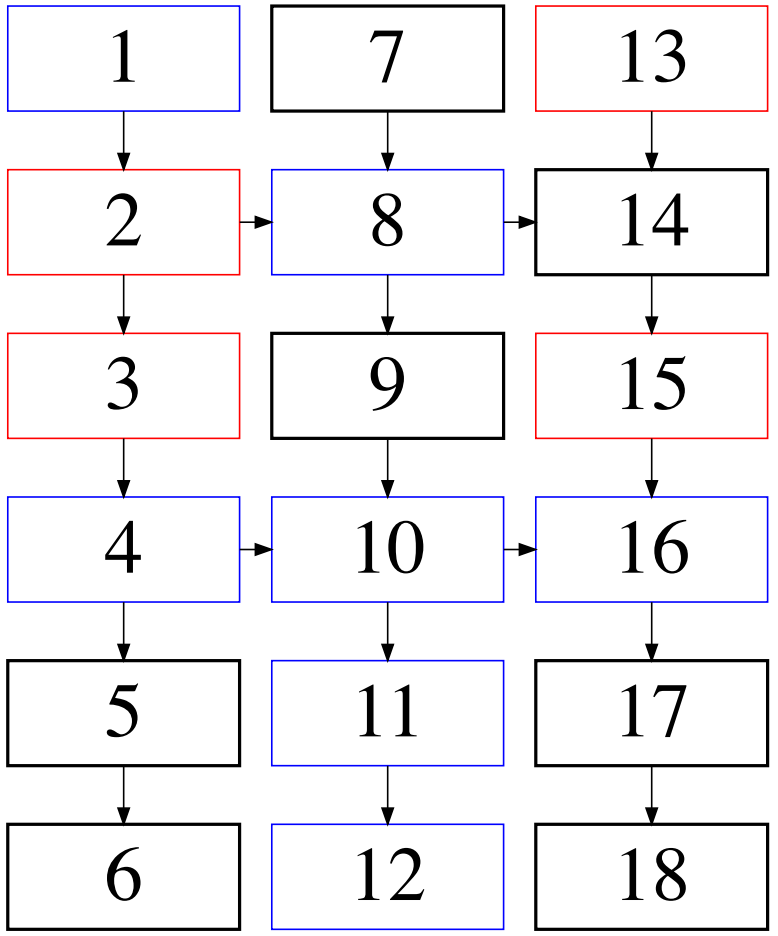
Attempt 2 - Define Grid

```
grViz("
digraph {

#First column
1 [style = 'solid', shape = 'box', width = 2, fontsi
2 [style = 'solid', shape = 'box', width = 2, fontsi
3 [style = 'solid', shape = 'box', width = 2, fontsi
4 [style = 'solid', shape = 'box', width = 2, fontsi
5 [style = 'solid', shape = 'box', width = 2, fontsi
6 [style = 'solid', shape = 'box', width = 2, fontsi

#Second column
7 [style = 'solid', shape = 'box', width = 2, fontsi
8 [style = 'solid', shape = 'box', width = 2, fontsi
9 [style = 'solid', shape = 'box', width = 2, fontsi
10 [style = 'solid', shape = 'box', width = 2, fonts
11 [style = 'solid', shape = 'box', width = 2, fonts
12 [style = 'solid', shape = 'box', width = 2, fonts

#Third column
13 [style = 'solid', shape = 'box', width = 2, fonts
14 [style = 'solid', shape = 'box', width = 2, fonts
```



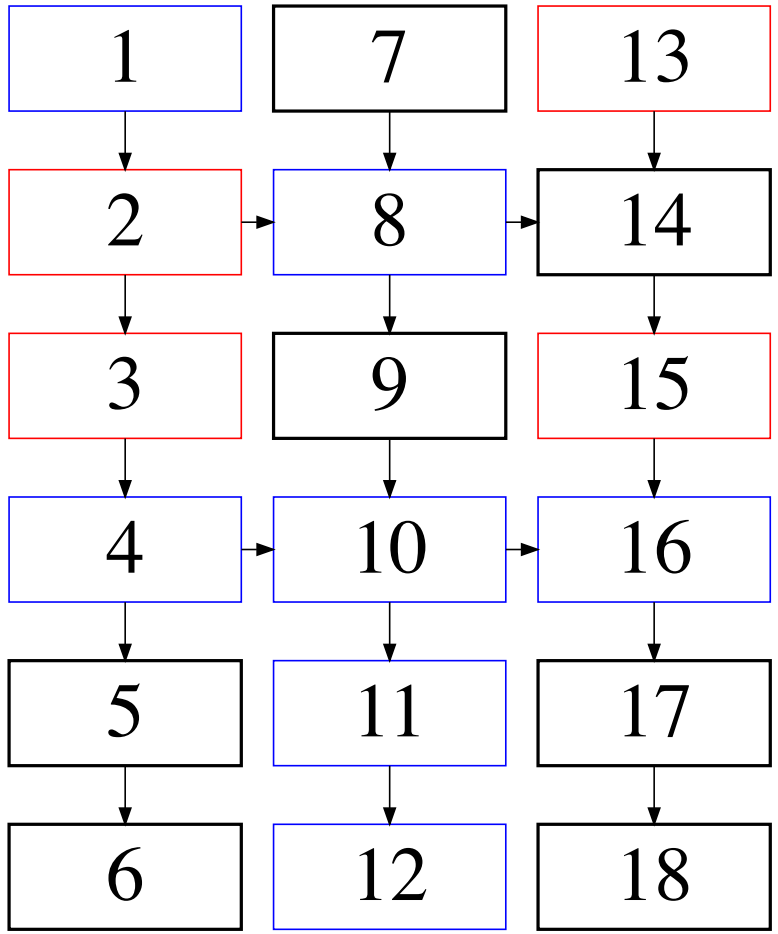
Attempt 2 - Define Grid

```
grViz("
digraph {

#First column
1 [style = 'solid', shape = 'box', width = 2, fontsi
2 [style = 'solid', shape = 'box', width = 2, fontsi
3 [style = 'solid', shape = 'box', width = 2, fontsi
4 [style = 'solid', shape = 'box', width = 2, fontsi
5 [style = 'solid', shape = 'box', width = 2, fontsi
6 [style = 'solid', shape = 'box', width = 2, fontsi

#Second column
7 [style = 'solid', shape = 'box', width = 2, fontsi
8 [style = 'solid', shape = 'box', width = 2, fontsi
9 [style = 'solid', shape = 'box', width = 2, fontsi
10 [style = 'solid', shape = 'box', width = 2, fonts
11 [style = 'solid', shape = 'box', width = 2, fonts
12 [style = 'solid', shape = 'box', width = 2, fonts

#Third column
13 [style = 'solid', shape = 'box', width = 2, fonts
14 [style = 'solid', shape = 'box', width = 2, fonts
```



Attempt 2 - Define Grid

- Red: Will be removed from the visual
- Blue: Will be modified; here, we will make the nodes and arrowheads essentially invisible; we will also add some headings
- Black: Keep for visual; they are the backbone of the visual

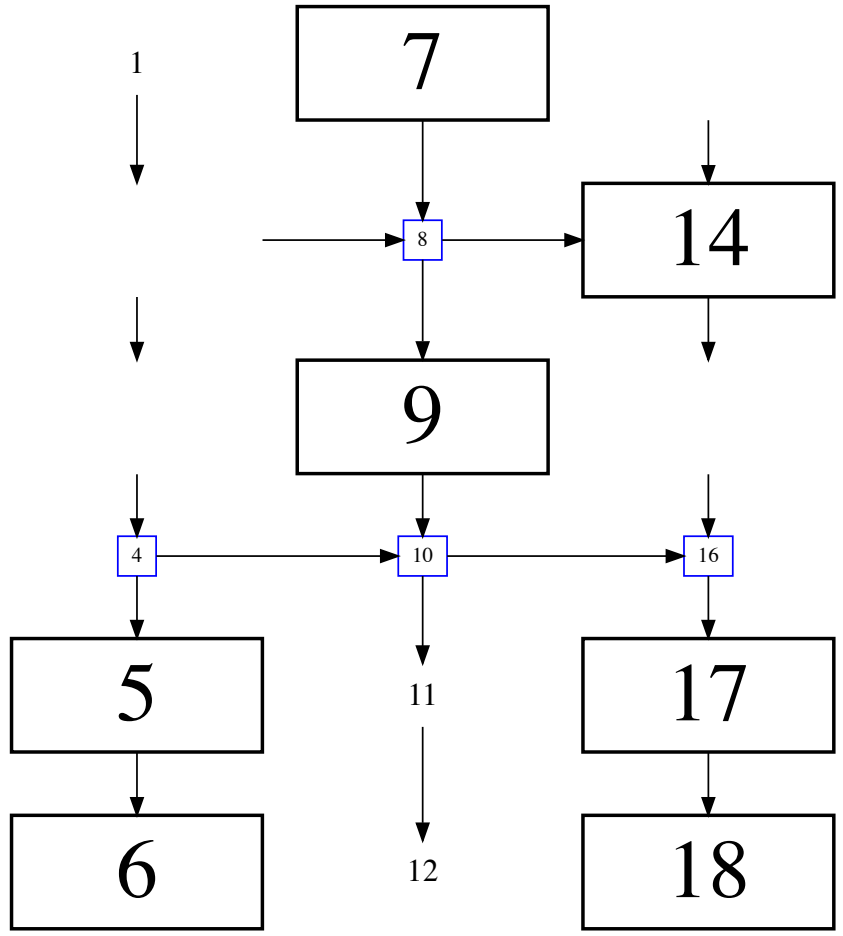


Attempt 2 - Shape the Diagram

- `shape = 'none'`: Remove outlines of nodes

- `height = 0.1, width = 0.1`: Started shrinking blue nodes; they will be a size of 0 for next step
  - Notice that the arrows are starting to converge together in those locations

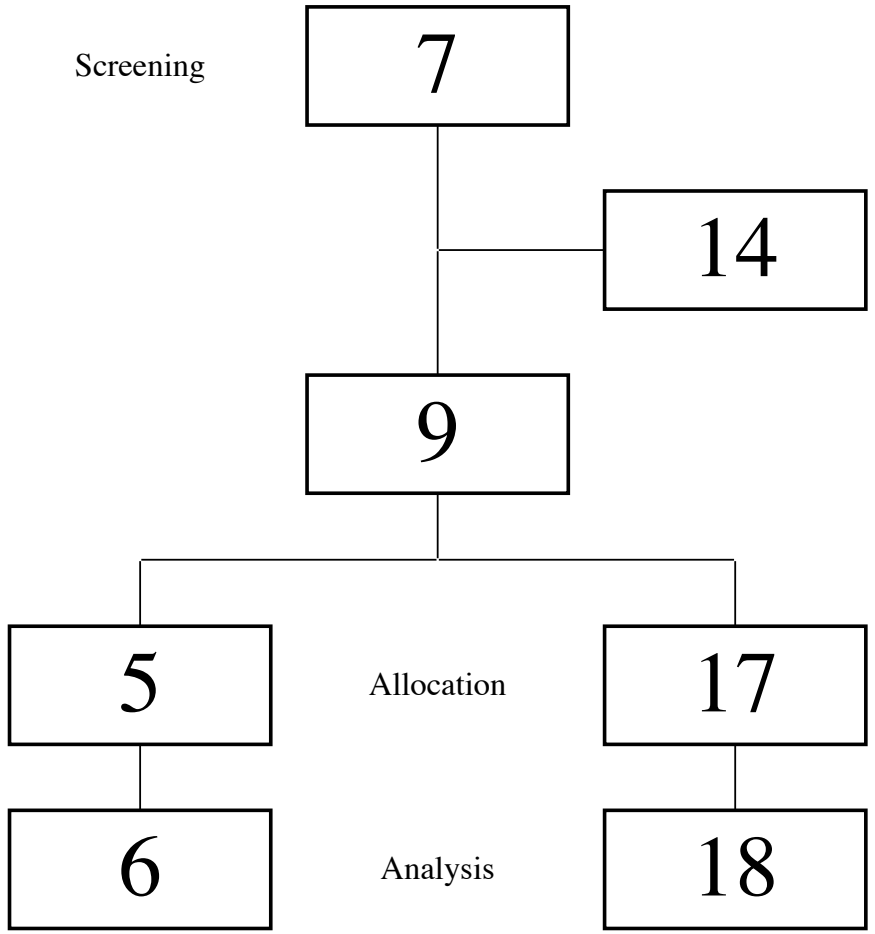
- `fontsize = 12`: Adjusted font size for the headings



Attempt 2 - Finalize the Diagram

- `label = ''`: Make the labels invisible in order to connect various edges
  - `4 [label = '', style = 'invisible', shape = 'none', height = 0, width = 0, fontsize = 12, color = blue]`

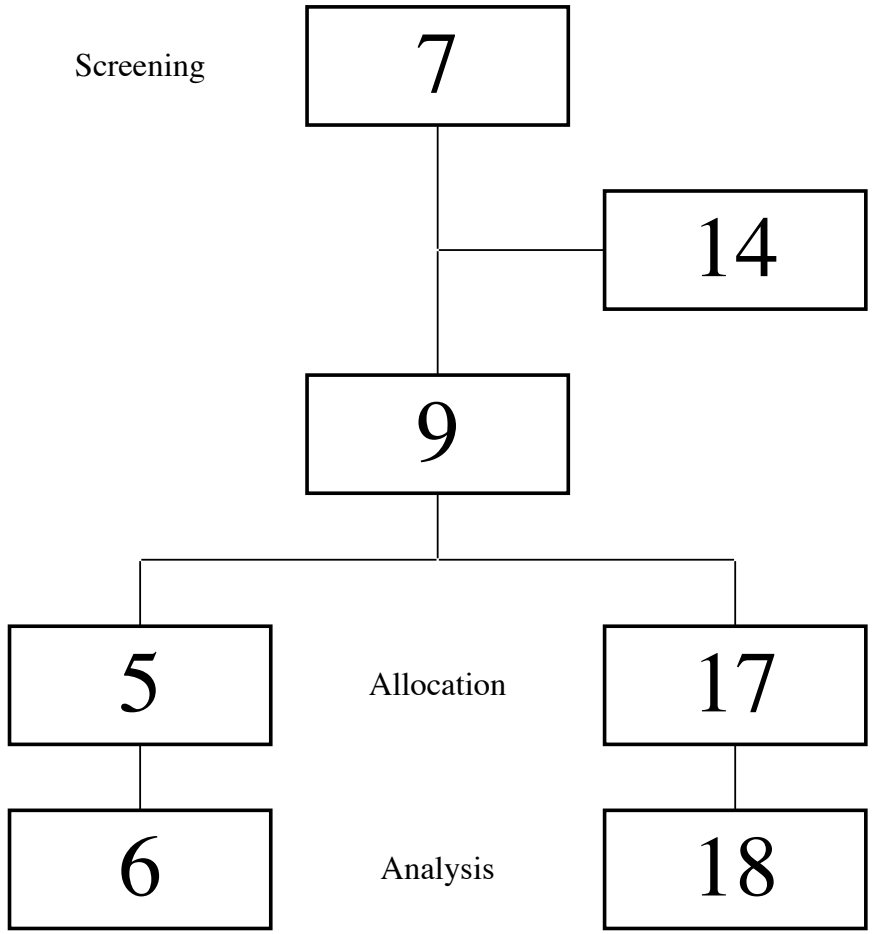
- `[arrowhead = 'none', color = black]`: Remove arrowheads, but keep as black
  - Use `color = #00000000` to make them invisible



Attempt 2 - Finalize the Diagram

- `graph [nodesep = 0.28, ranksep = 0.5]`: Adjust the vertical and horizontal space between the nodes

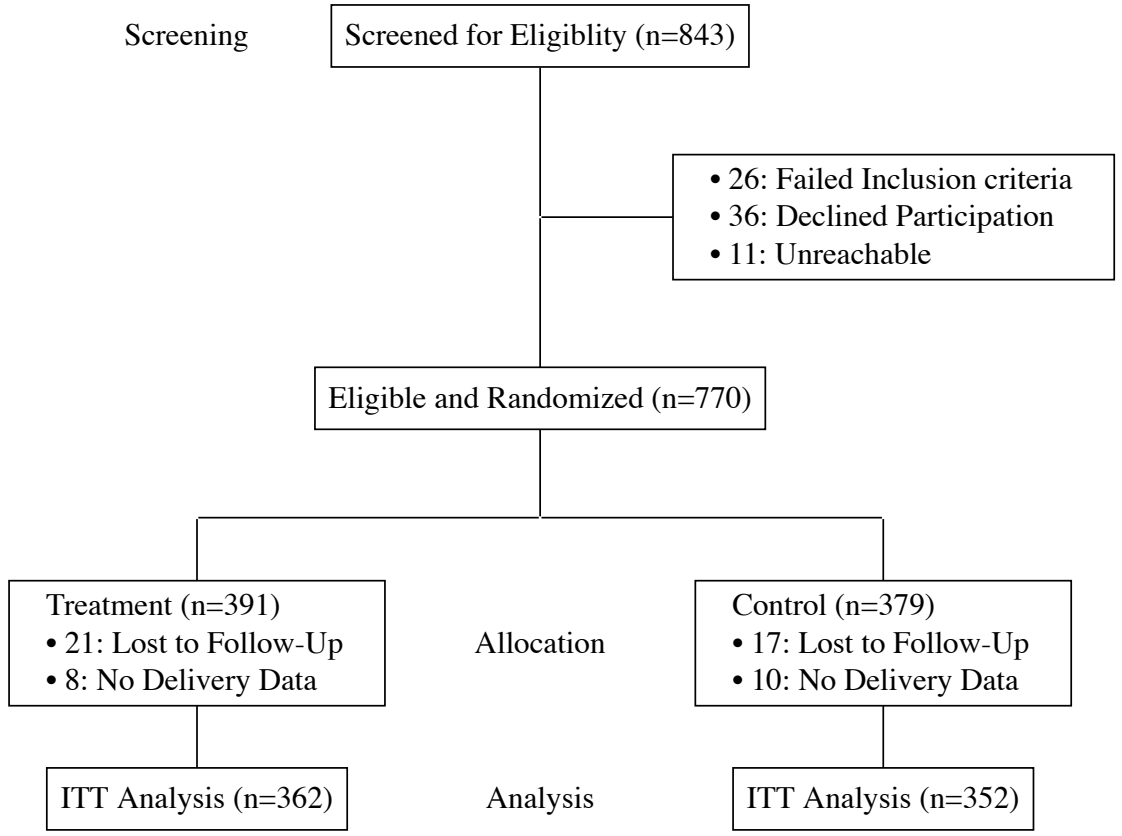
- `label = 'Screening'`: Change the labels of each node



Attempt 2 - Finish the Diagram

- `graph [nodesep = 0.28, ranksep = 0.5]`: Adjust the vertical and hortizontal space between the nodes

- `label = 'Screening'`: Change the labels of each node



Export Diagram

- To export the diagram, use `library(rsvg)` and `library(DiagrammeRsvg)`

- Assign `grViz` output into an object and run `export_svg %>% charToRaw %>% rsvg_pdf("consort_diagram.pdf")`

- Replace `rsvg_pdf` with `rsvg_png` to save as PNG file instead

Final Code

```
grViz("
digraph {

#First column
1 [label = 'Screening', style = 'solid', shape = 'none', width = 2, fontsize = 18, color = black]
2 [style = 'invisible', shape = 'box', width = 2, fontsize = 48, color = red]
3 [style = 'invisible', shape = 'box', width = 2, fontsize = 48, color = red]
4 [label = '', style = 'invisible', shape = 'none', height = 0, width = 0, fontsize = 12, color = b
5 [label = <
    Treatment (n=391)<br ALIGN = 'LEFT'/>
    &#8226; 21: Lost to Follow-Up<br ALIGN = 'LEFT'/>
    &#8226; 8: No Delivery Data<br ALIGN = 'LEFT'/>>,
    style = 'solid', shape = 'box', width = 2, fontsize = 18, color = black, penwidth = 1]
6 [label = 'ITT Analysis (n=362)', style = 'solid', shape = 'box', width = 2, fontsize = 18, color

#Second column
7 [label = 'Screened for Eligiblity (n=843)', style = 'solid', shape = 'box', width = 2, fontsize =
8 [label = '', style = 'invisible', shape = 'none', height = 0, width = 0, fontsize = 12, color = b
9 [label = 'Eligible and Randomized (n=770)', style = 'solid', shape = 'box', width = 2, fontsize =
10 [label = '', style = 'invisible', shape = 'none', height = 0, width = 0, fontsize = 12, color =
11 [label = 'Allocation', style = 'solid', shape = 'none', width = 2, fontsize = '18', color = blue
12 [label = 'Analysis', style = 'solid', shape = 'none', width = 2, fontsize = '18', color = blue]
```

Final Code (Cont.)

```
grViz("
digraph {

#Third column
13 [style = 'invisible', shape = 'box', width = 2, fontsize = 48, color = red] #BREAK4
14 [label = <
    &#8226; 26: Failed Inclusion criteria<br ALIGN = 'LEFT'/>
    &#8226; 36: Declined Participation<br ALIGN = 'LEFT'/>
    &#8226; 11: Unreachable<br ALIGN = 'LEFT'/>>,
    style = 'solid', shape = 'box', width = 2, fontsize = 18, color = black, penwidth = 1]
15 [style = 'invisible', shape = 'box', width = 2, fontsize = 48, color = red] #BREAK4
16 [label = '', style = 'invisible', shape = 'none', height = 0, width = 0, fontsize = 12, color =
17 [label = <
    Control (n=379)<br ALIGN = 'LEFT'/>
    &#8226; 17: Lost to Follow-Up<br ALIGN = 'LEFT'/>
    &#8226; 10: No Delivery Data<br ALIGN = 'LEFT'/>>,
    style = 'solid', shape = 'box', width = 2, fontsize = 18, color = black, penwidth = 1]
18 [label = 'ITT Analysis (n=352)', style = 'solid', shape = 'box', width = 2, fontsize = 18, color

}"
)
```

Final Code (Cont.)

```
grViz("
digraph {

#Add edges
edge [arrowhead = 'normal', color = '#00000000']
1 -> 2; 2 -> 3; 3 -> 4; 10 -> 11; 11 -> 12; 13 -> 14; 14 -> 15; 15 -> 16

edge [arrowhead = 'none', color = black]
4 -> 5; 5 -> 6; 7 -> 8; 8 -> 9; 9 -> 10; 16 -> 17; 17 -> 18

#Add additional arrowheads
edge [arrowhead = 'none', color = '#00000000']
2 -> 8

edge [arrowhead = 'none', color = 'black']
8 -> 14; 4 -> 10; 10 -> 16

#Use rank to keep these three boxes on the same rank/level!!!
{rank=same; '2';'8';'14'} {rank=same; '4';'10';'16'}

#Use nodesep and ranksep to keep the nodes a bit closer together
graph [nodesep = 0.10, ranksep = 0.5]
```

It’s so beautiful.

Summary and Notes

- CONSORT diagrams provide an effective way of showing participant flow
- We used the DiagrammeR package in R and worked with the GraphViz functionality to build the diagram
    - We did not use the native built-in approach - Syntax is also generally straightforward
- Graph specifications consist of node, edge, and graph statements
- When creating CONSORT diagrams, visualize the grid first!
    - Try to anticipate what nodes you need, what will be modified, and what can go
- Don't limit yourself to just use for randomized trials!

Thank You!!!