MMU®
MULTIMEDIA UNIVERSITY

TPT1201
Research Methodology

Lab 8

By completing the tutorial, you should be able to:

- create a simple flowchart using LaTeX
- add colours to the flowchart components

Exercise 1

In this exercise, we're going to be looking at creating flowcharts in TikZ. To get started we need to load up the `tikz` package, the `shapes.geometric` TikZ library and the arrows library.

```
\usepackage{tikz}
\usetikzlibrary{shapes.geometric, arrows}
```

Exercise 2

The `tikzstyle` command

Now before we start the document we need to define the basic components of a flowchart. To do this we use the `tikzstyle` command. First, let's define the block we're going to use for start and stop blocks. We'll name it *startstop* using curly brackets immediately following the command, then we add an equals sign before a set of square brackets. In the square brackets we enter all the formatting information. For this block we'll specify a rectangle with rounded corners. We'll give it a minimum width of 3cm and a minimum height of 1cm. We'll also ensure the text gets centred and we'll set both a draw and a fill colour. In this example we've set the fill colour to a colour that is 30% red mixed with 70% white:

```
\tikzstyle{startstop} = [rectangle, rounded corners,
minimum width=3cm, minimum height=1cm,
text centered, draw=black, fill=red!30]
```

Exercise 3

Next we'll specify an *input or output* box. This time we want the block to be a parallelogram. To achieve this we ask for a trapezium and then alter the angles. The rest is very similar:

```
\tikzstyle{io} = [trapezium, trapezium left angle=70,
trapezium right angle=110, minimum width=3cm,
minimum height=1cm, text centered, draw=black,
fill=blue!30]
```

Exercise 4

Next, we create *process* and *decision* nodes.

```
\tikzstyle{process} = [rectangle, minimum width=3cm,
minimum height=1cm, text centered, draw=black,
fill=orange!30]

\tikzstyle{decision} = [diamond, minimum width=3cm,
minimum height=1cm, text centered, draw=black,
fill=green!30]
```

Exercise 5

Finally we'll define a style for the arrows. For this we set the line thickness to thick, add an arrow head and specify the stealth arrow head:

```
\tikzstyle{arrow} = [thick,->,>=stealth]
```

Exercise 6

Now we are ready to start building our flowchart. To do the we use the tikzpicture environment. We'll create our flowchart blocks using nodes and the tikzstyles we defined earlier. The nodes we use to build the blocks are automatically spaced 2cm apart from their centres:

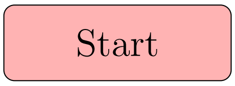
```
\begin{tikzpicture}[node distance=2cm]
... your codes here...
\end{tikzpicture}
```

Exercise 7

Let's try the following code:

```
\begin{tikzpicture}[node distance=2cm]
\node (start) [startstop] {Start};
\end{tikzpicture}
```

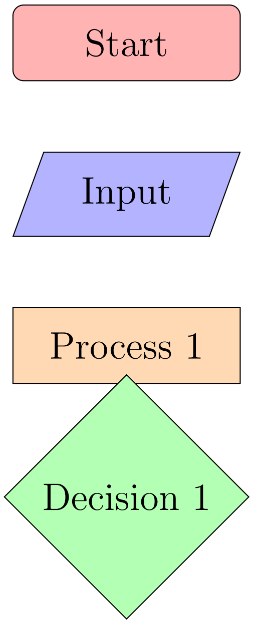
You should see the node named "Start" below:



Exercise 8

Add the following lines into the codes in the previous exercise:

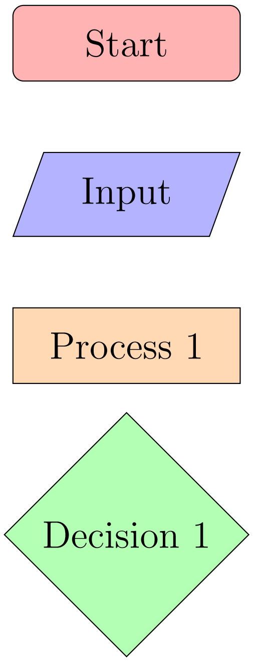
```
\begin{tikzpicture}[node distance=2cm]
\node (start) [startstop] {Start};
\node (in1) [io, below of=start] {Input};
\node (pro1) [process, below of=in1] {Process 1};
\node (dec1) [decision, below of=pro1] {Decision 1};
\end{tikzpicture}
```



Exercise 9

Do you notice that the node “Decision 1” is touching “Process 1”? To solve this problem, we need to shift the “Decision 1” node down a bit.

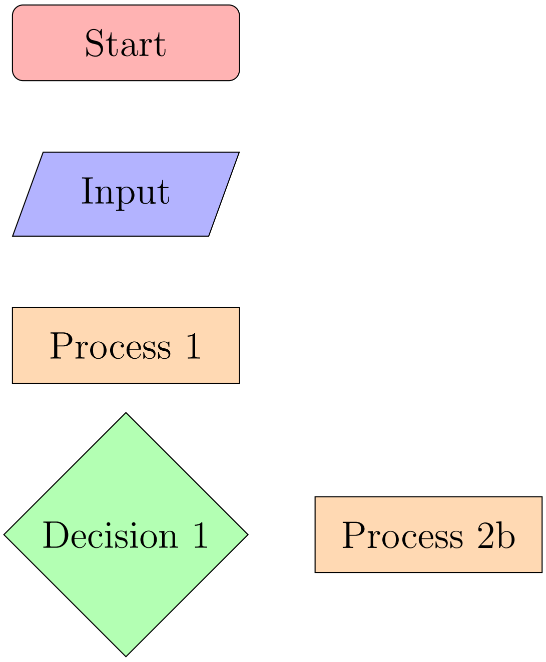
```
\node (dec1) [decision, below of=pro1,
    yshift=-0.5cm] {Decision 1};
```



Exercise 10

Let's add one *process* node to the right of *decision* node, by adding the following code:

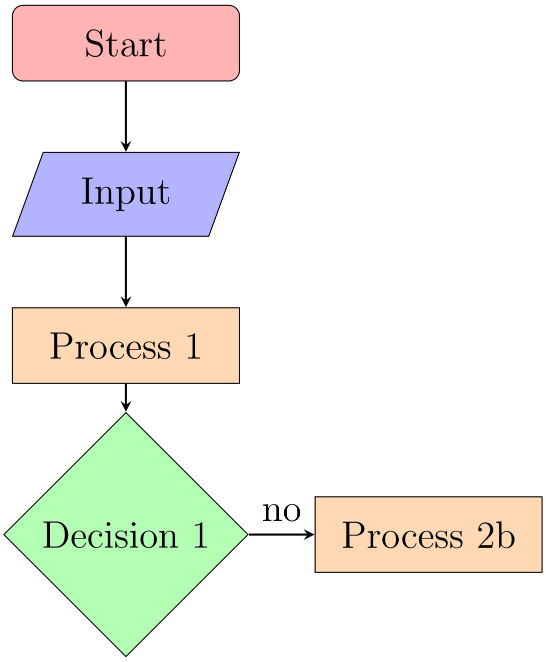
```
\node (pro2b) [process, right of=dec1,
    xshift=2cm] {Process 2b};
```



Exercise 11

Let's add arrows to the code in Exercise 10:

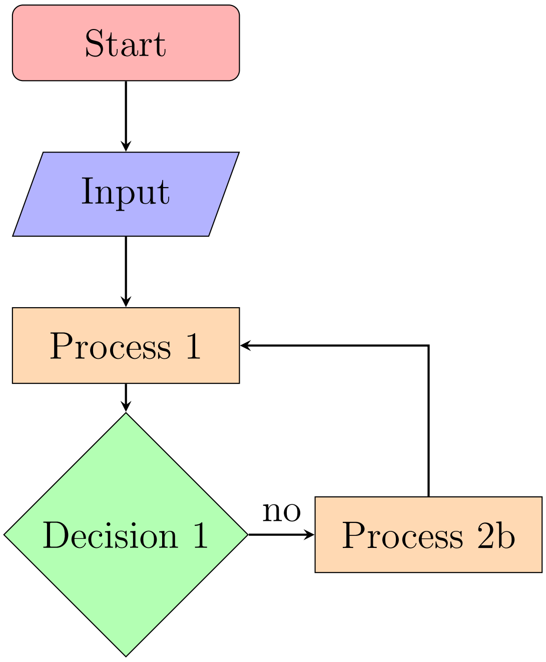
```
\draw [arrow] (start) -- (in1);
\draw [arrow] (in1) -- (pro1);
\draw [arrow] (pro1) -- (dec1);
\draw [arrow] (dec1) -- (pro2b);
```



Exercise 12

Let's us add one line from "Process 2b" to "Process 1".

```
\draw [arrow] (pro2b) |- (pro1);
```


Start
Input
Process 1
Decision 1
no
Process 2b