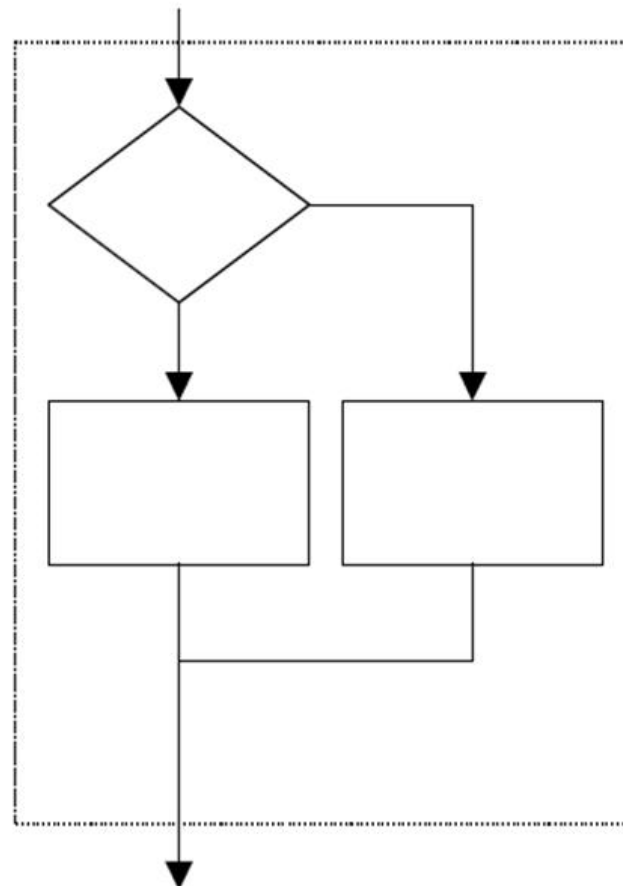


Branches

Statements that permit us to select and execute specific sections of code while skipping other sections (in MATLAB):

- *if*
- *switch*
- *try /catch*



The *if* construct

```
If control_expr  
    statement1  
    statement 2  
    ...  
end
```

```
if Temp_F<45  
    disp('esta haciendo un frio peluuu !!!!')  
end
```

The *if* construct

```
If control_expr
    statements
else
    statements
end
```

```
if Temp_F<45
    disp('esta haciendo un frio peluuu !!!!')
else
    disp('esta chillin')
end
```

The *if* construct

```
If control_expr
    statements
elseif
    statements
else
    statements
end
```

```
if Temp_F<45
    disp('esta haciendo un frio peluuu !!!!')
elseif Temp_F>90
    disp('esta haciendo un calor brutal !!!!')
else
    disp('esta chillin')
end
```

The *if* construct

- You must have one *end* statement for each *if* statement
- You can have any number of *else if* clauses
- You may (or not) have one *else* clause
- If constructs may be nested:

```
If control_expr1
    statements
    if control_expr2
        statements
    end
end
```

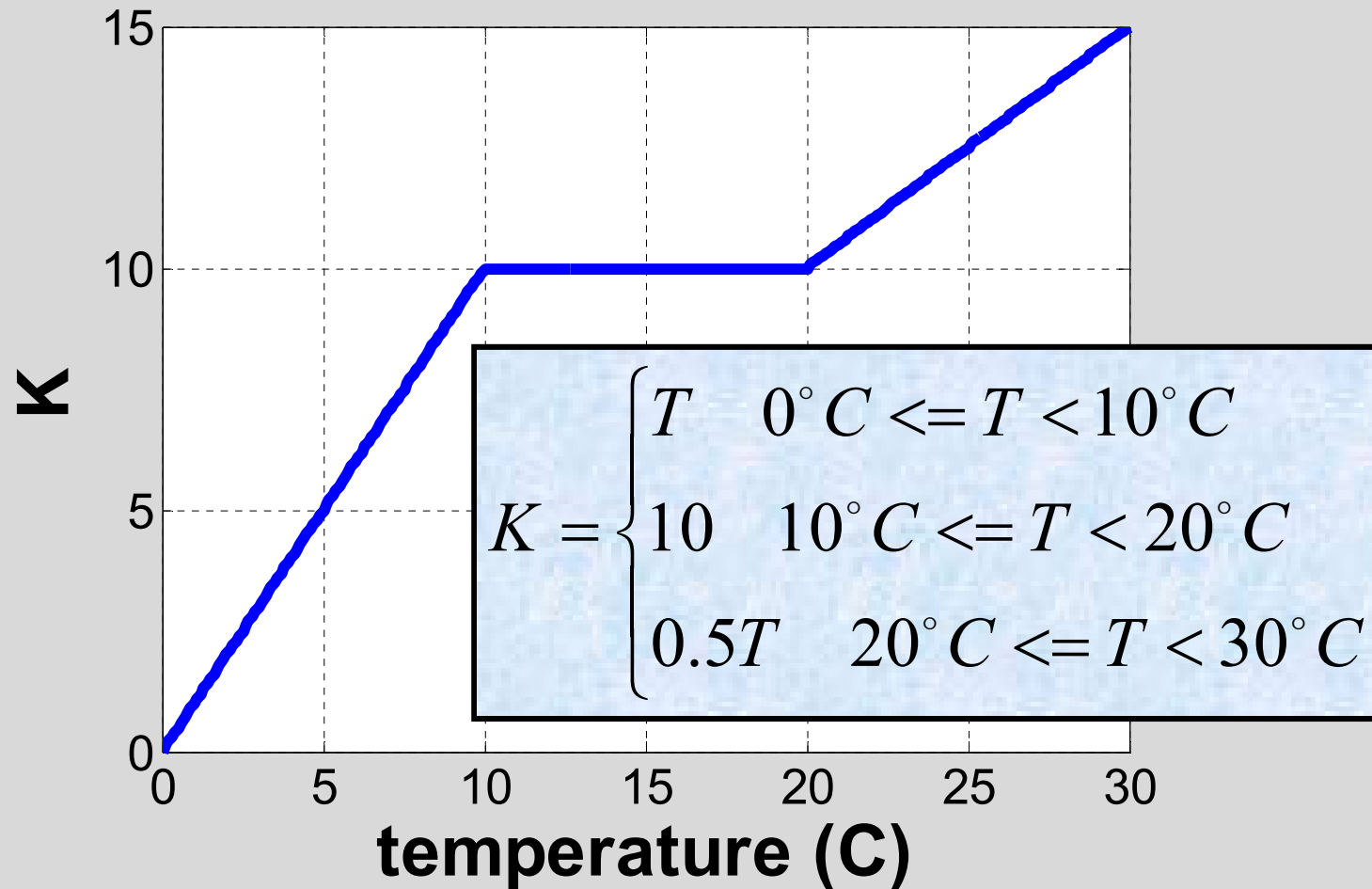
The *if* construct

```
if Temp_F<45
    disp('esta haciendo un frio peluuu !!!!')
elseif Temp_F>=45 & Temp_F<65
    disp('esta friito mano !!!!')
elseif Temp_F>=65 & Temp_F<90
    disp('esta chillin')
elseif Temp_F>=90
    disp('esta haciendo un calor brutal !!!!')
end
```

The *if* construct

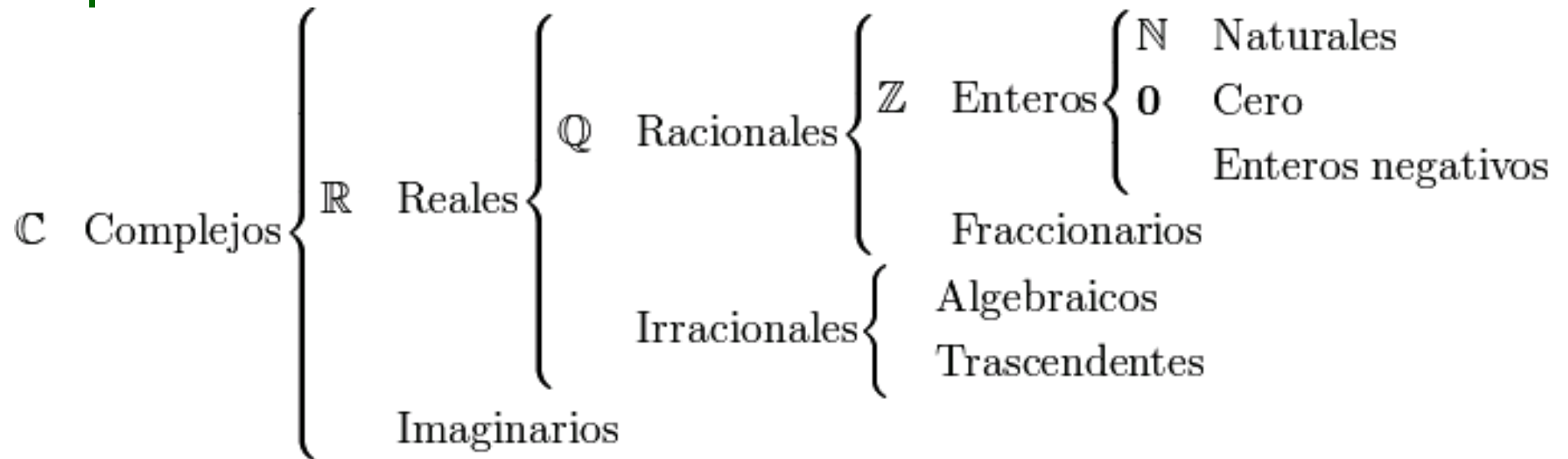
```
if Temp_F >= 45
    if Temp_F < 65
        disp('esta friito mano !!!!')
    else
        if Temp_F < 90
            disp('esta chillin !!!!')
        else
            disp('esta haciendo un calor brutal !!!!')
        end
    end
else
    disp('esta haciendo un frio peluuu !!!!')
end
```

La constante K de un gas depende de la temperatura tal como lo muestra la siguiente grafica:



Escriba un programa que lea la temperatura y calcule la variable K.

Recordando la escuela elemental



1. Escriba un programa que lea un número real y decida si es entero o no.
2. Escriba una función que se llame "*esreal*" que reciba un número y devuelva un 1 si es entero, un 0 si no.

```
clc, clear, close all
```

```
% Este programa lee un numero real y decide si es entero o no
```

```
=====
```

```
numero = input('Entre un numero real: ');
```

```
imag_part = imag(numero);
```

```
% parte imaginaria del numero
```

```
if imag_part==0
```

```
    decimal_part = numero-fix(numero); % parte decimal del numero
```

```
    if decimal_part==0
```

```
        disp([num2str(numero),' es un numero entero']);
```

```
    else
```

```
        disp([num2str(numero),' no es un numero entero']);
```

```
    end
```

```
else
```

```
    disp([num2str(numero),' ???? te regalaron la high!!!']);
```

```
end
```

Escriba un programa que le pida al usuario los coeficientes ***a***, ***b*** y ***c*** de una ecuación cuadrática. El programa deberá decirle al usuario cuantas raíces y de qué tipo tiene la ecuación, y calcular dichas raíces. Valide que los coeficientes que entra el usuario sean números reales.

```

Command Window

=====
===== calcula las raices de una ecuacion cuadratica =====
=====          ax^2 + bx + c = 0          =====
=====

entre un valor para el coeficiente a : 2
entre un valor para el coeficiente b : 3
entre un valor para el coeficiente c : 1

dos raices reales distintas x1 = -0.5 y x2 = -1

=====

chequeo de las raices :
f(x1) ----> 0
f(x1) ----> 0
=====
fx >> |

```

Escriba ahora el programa compuesto de funciones