

MEMORIAS-completa-10.pdf



paulaaaaa13



Informática Aplicada a la Bioquímica



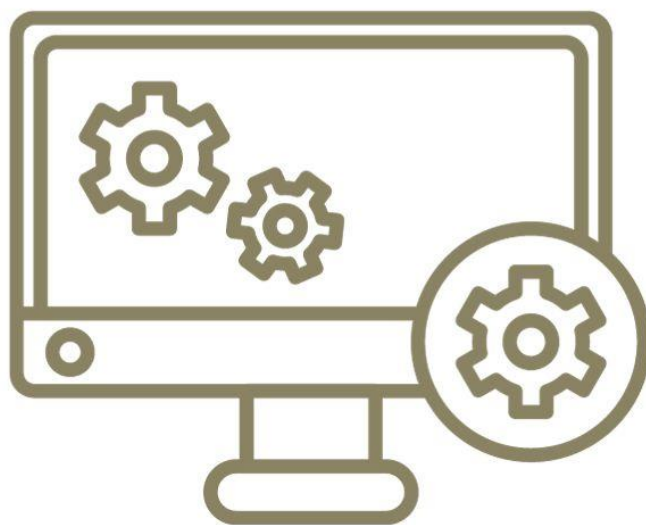
2º Grado en Bioquímica



**Facultad de Ciencias
Universidad de Córdoba**

cuaderno de prácticas
INFORMATICA
aplicada a la bioquímica

paula Ferrer Ordóñez



B22FEORP@UCO.ES

WUOLAH

Índice

PRÁCTICA 1: LINUX	3
Parte 1. Introducción a Linux	3
Parte 2.....	5
PRACTICA 2: HOJA DE CALCULO.....	8
Indica las diferencias/mejoras que aporta Excel en Office 365 vs Google Sheets. ..	19
PRACTICA 3. EDICIÓN DE TEXTOS CIENTÍFICOS CON LATEX.....	21
Copiar el ejemplo modelo.....	21
PRACTICA 4. BASES DE DATOS BIBLIOGRÁFICA.....	24
ENDNOTE.	25
MENDELEY	27
GOOGLE SCHOLAR	29
PRACTICA 5 : INTRODUCCIÓN A R STUDIO	31
Ejercicios parte 1.	35
Ejercicios parte 2.	47
Ejercicios Parte 3.....	53
Ejercicios Parte 4.....	63
Ejercicios parte 5.	71
Ejercicios parte 6	78
Ejercicios parte 7.	83

PRÁCTICA 1: LINUX

Parte 1. Introducción a Linux

En esta parte de la practica se aprende la utilización de comandos básicos de la terminal de Linux además de aprender algunas notas útiles de cómo funcionan estos comandos. A continuación, una lista de los comandos y sus respectivos usos:

- **pwd.** Para saber en que directorio estás.
- **ls.** Para ver los contenidos de un directorio. Para ver un resultado más completo y en forma de lista se utiliza `ls -lag`
- **mkdir.** Para crear un directorio. (espacio nombre del directorio que quieres crear)
- **cd.** Para cambiar de directorio (espacio nombre del directorio que quieres abrir) (si solo pones `cd` te dirige al directorio principal)
- **cd ..** Para dirigirte al directorio anterior al que estás.
- **rmdir.** Para borrar un directorio vacío (espacio nombre del directorio)
- **gedit.** Para abrir un editor de textos
- **cat.** Te muestra el contenido de un archivo de texto (espacio nombre del archivo de texto)
- **more.** Igual que `cat` pero con archivos más grandes.
- **less.** Igual que `cat` pero para archivos más grandes.
- **cp.** Para copiar un archivo (espacio nombre del archivo que queremos copiar espacio nombre del archivo al que copiamos)
- **mv.** Mover el archivo a otro directorio (espacio nombre del archivo espacio directorio al que lo queramos mover)
- **rm.** Borrar archivo (espacio nombre del archivo)
- **man.** Nuestra información del comando (espacio comando)
- **wc.** Contar caracteres

Hay que tener en cuenta que la terminal es sensible al uso de minúsculas y mayúsculas y que para introducir un comando hay que pulsar `intro`.

Para borrar todos los archivos de un mismo tipo= `rm *.mp3` donde `*` sustituye a 0 o más caracteres (ejemplo `rm *2*` borraría todos los archivos que contengan un 2 en su nombre)

`rm` no se puede recuperar

Ejercicio 1.

```
b22feorp@AAVA1015:~$ cd curso2324
```

```
b22feorp@AAVA1015:~/curso2324$ mkdir IAB
```

```

b22feorp@AAVA1015:~/curso2324$ cd IAB
b22feorp@AAVA1015:~/curso2324/IAB$ mkdir cuaderno
b22feorp@AAVA1015:~/curso2324/IAB$ cd cuaderno
b22feorp@AAVA1015:~/curso2324/IAB/cuaderno$ mkdir p0
b22feorp@AAVA1015:~/curso2324/IAB/cuaderno$ mkdir p1
b22feorp@AAVA1015:~/curso2324/IAB/cuaderno$ mkdir p2
b22feorp@AAVA1015:~/curso2324/IAB/cuaderno$ mkdir p3
b22feorp@AAVA1015:~/curso2324/IAB/cuaderno$ mkdir p4
b22feorp@AAVA1015:~/curso2324/IAB/cuaderno$ ls
p0 p1 p2 p3 p4

```

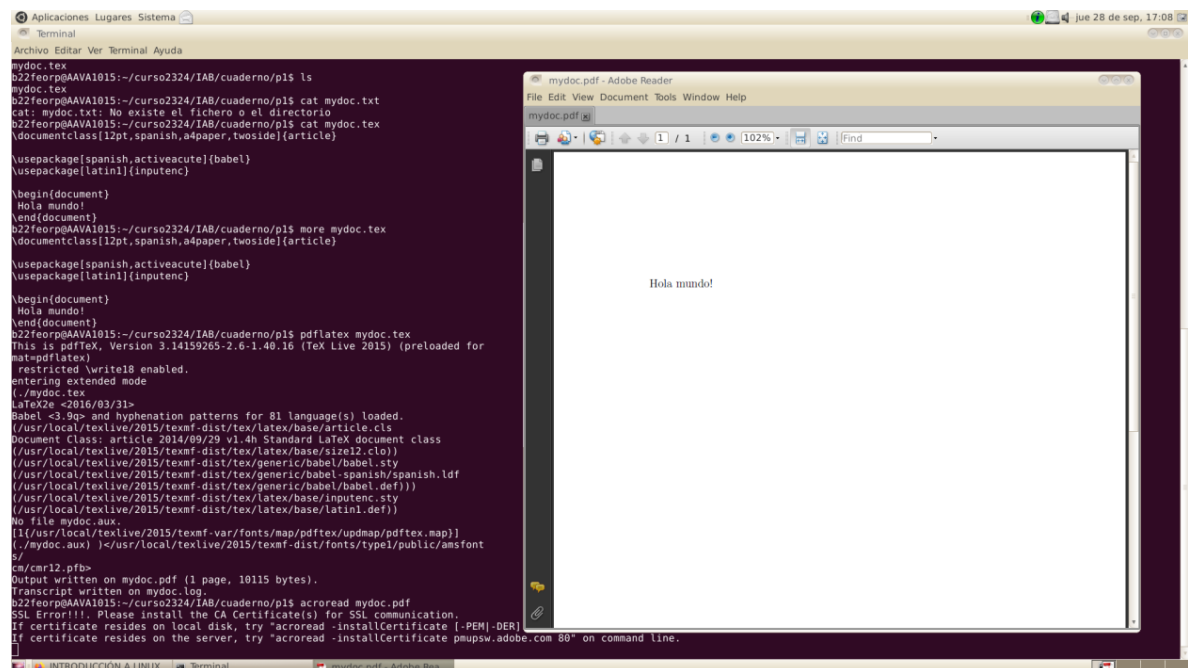
Ejercicio 2.

```

b22feorp@AAVA1015:~/curso2324/IAB/cuaderno/p1$ ls
mydoc.tex

```

Ejercicio 3.



Ejercicio 4

b22feorp@AAVA1015:~\$ firefox &

[2] 19258

[1] Hecho firefox

Parte 2

En este caso el objetivo de esta práctica es aprender a como trabajar con un fichero de texto en formato FASTA, en este caso contiene una secuencia de ADN de e. coli

Ejercicios 1 y 2.

b22feorp@AAVA1015:~/Descargas\$ more sequence.fasta

>AY893111.1 Synthetic construct Homo sapiens clone FLH119406.01L alkB, alkylatio

n repair-like (E. coli) mRNA, partial cds

ATGGGGAAGATGGCAGCGCCGTGGGCTCTGTGGCGACTCTGGCGACTGAGCCCGGGGAGGACGCCT
TTC

GGAAACTTTTCCGCTTCTACCGTCAGAGCCGGCCCGGGACCGCAGACCTGGAAGGGGTCATCGACTTCT
C

GGCGGCCACGCAGCCCGTGGCAAGGGTCTGGTGCCCAAAAGGTGATCAAATCTCAGCTAAATGTGTC
T

TCTGTCAGTGAGCAGAATGCATATAGAGCAGGTCTTCAGCCCGTCAGCAAGTGGCAAGCCTATGGACTCA

AAGGCTATCCTGGGTTTATTTTATCCCAAACCCCTTCCTCCAGGTTACCACTGGCACTGGGTGAAACA

GTGCCTTAAGTTATATTCCAGAAACCTAATGTATGTAACCTGGACAAACACATGTCTAAAGAAGAGACC

CAAGATCTGTGGGAACAGAGCAAAGAGTTCCTGAGGTATAAAGAAGCGACTAAACGGAGACCCCGAAG
TT

TACTGGAGAACTGCGTTGGGTGACCGTAGGCTACCATTATAACTGGGACAGTAAGAAATACTCAGCAGA

TCATTACACACCTTTCCCTTCTGACCTGGGTTTCCTCTCAGAGCAAGTAGCCGCTGCCTGTGGATTTGAG

GATTTCCGAGCTGAAGCAGGGATCCTGAATTACTACCGCCTGGACTCCACACTGGGAATCCACGTAGACA

GATCTGAGCTAGATCACTCCAAACCCCTTGCTGTCATTAGCTTTGGACAGTCCGCCATCTTTCTCCTGGG

TGGTCTTCAAAGGGATGAGGCCCCACGGCCATGTTTATGCACAGTGGTGACATCATGATAATGTCGGGT

TTCAGCCGCCTCTTGAACCACGCAGTCCCTCGTGTCTTCCAAATCCAGAAGGGGAAGGCCTGCCTCACT
GCCTAGAGGCACCTCTCCCTGCTGTCTCCCGAGAGATTCAATGGTAGAGCCTTGTTCTATGGAGGACTG
GCAGGTGTGTGCCAGCTACTTGAAGACCGCTCGTGTTAATGACTGTCCGACAGGTCCTGGCCACAGAC
CAGAATTTCCCTCTAGAACCCATCGAGGATGAAAAAAGAGACATCAGTACAGAAGGTTTCTGCCATCTGG
ATGACCAGAATAGCGAAGTAAACGGGCCAGGATAAACCCCTCACAGCTTG

b22feorp@AAVA1015:~/Descargas\$ nano sequence.fasta

b22feorp@AAVA1015:~/Descargas\$ more sequence.fasta

ATGGGGAAGATGGCAGCGCCGTGGGCTCTGTGGCGACTCTGGCGACTGAGCCCGGGGAGGACGCCT
TTC
GGAAACTTTTCCGCTTCTACCGTCAGAGCCGGCCCGGGACCGCAGACCTGGAAGGGGTCATCGACTTCT
C
GGCGGCCACGCAGCCCGTGGCAAGGGTCCTGGTGCCAAAAGGTGATCAAATCTCAGCTAAATGTGTC
T
TCTGTCA GTGAGCAGAATGCATATAGAGCAGGTCTTCAGCCCGTCAGCAAGTGGCAAGCCTATGGACTCA
AAGGCTATCCTGGGTTTATTTTATCCCAAACCCCTTCTCCCAGGTTACCAGTGGCACTGGGTGAAACA
GTGCCTTAAGTTATATTCCAGAAACCTAATGTATGTAACCTGGACAAACACATGTCTAAAGAAGAGACC
CAAGATCTGTGGGAACAGAGCAAAGAGTTCCTGAGGTATAAAGAAGCGACTAAACGGAGACCCCGAAG
TT
TACTGGAGAACTGCGTTGGGTGACCGTAGGCTACCATTATAACTGGGACAGTAAGAAATACTCAGCAGA
TCATTACACACCTTTCCCTTCTGACCTGGGTTTCTCTCAGAGCAAGTAGCCGCTGCCTGTGGATTTGAG
GATTTCCGAGCTGAAGCAGGGATCCTGAATTACTACCGCCTGGACTCCACACTGGGAATCCACGTAGACA
GATCTGAGCTAGATCACTCCAAACCCCTTGCTGTCAATCAGCTTTGGACAGTCCGCCATCTTCTCCTGGG
TGGTCTTCAAAGGGATGAGGCCCCACGGCCATGTTTATGCACAGTGGTGACATCATGATAATGTGGGT
TTCAGCCGCCTCTTGAACCACGCAGTCCCTCGTGTCTTCCAAATCCAGAAGGGGAAGGCCTGCCTCACT
GCCTAGAGGCACCTCTCCCTGCTGTCTCCCGAGAGATTCAATGGTAGAGCCTTGTTCTATGGAGGACTG
GCAGGTGTGTGCCAGCTACTTGAAGACCGCTCGTGTTAATGACTGTCCGACAGGTCCTGGCCACAGAC
CAGAATTTCCCTCTAGAACCCATCGAGGATGAAAAAAGAGACATCAGTACAGAAGGTTTCTGCCATCTGG
ATGACCAGAATAGCGAAGTAAACGGGCCAGGATAAACCCCTCACAGCTTG

Ejercicio 3.

```
b22feorp@AAVA1015:~/Descargas$ wc sequence.fasta
```

```
20 17 1190 sequence.fasta
```

Ejercicio 4.

```
<t sequence.fasta | tr -d -c '[:alnum:]' | wc -m
```

```
1170
```

En la pagina web sale el mismo número de caracteres que salen con el comando wc

Ejercicio 5.

```
b22feorp@AAVA1015:~/Descargas$ wc sequence.fasta
20 17 1190 sequence.fasta
<t sequence.fasta | tr -d -c '[:alnum:]' | wc -m
1170
b22feorp@AAVA1015:~/Descargas$ | tr -d -c '[:alnum:]' | wc -m
bash: error sintáctico cerca del elemento inesperado `|'
<as$ cat sequence.fasta | tr -d -c '[:alnum:]' | grep --color TAC
ATGGGGAAGATGGCAGCGGCCGTGGGCTCTGTGGCGACTCTGGCGACTGAGCCCGGGGAGGACGCCTTTCGGAACCTTT
CCGCTTCTACCGTCAGAGCCGGCCCGGGACCGCAGACCTGGAAGGGGTCATCGACTTCTCGCGGCCACGCAGCCCGTG
GCAAGGGTCCTGGTGCCCAAAGGTGATCAAATCTCAGCTAAATGTGTCTTCTGTCAAGTGAAGATGCATATAGAGCA
GGTCTTCAGCCCGTCAGCAAGTGGCAAGCCTATGGACTCAAAGGCTATCCTGGGTTTATTTTATCCCAAACCCCTTCCT
CCAGGTACAGTGGCACTGGGTGAAACAGTGCCTTAAGTTATATTCCAGAAACCTAATGTATGTAACCTGGACAAAC
ACATGTCTAAAGAAGAGACCCAAGATCTGTGGGAACAGAGCAAAGAGTTCTGAGGTATAAAGAAGCGACTAAACGGAGA
CCCCGAAGTTACTGGAGAACTGCGTTGGGTGACCGTAGGCACATTATAACTGGGACAGTAAGAAAACAGCAGAGA
TCATACACACCTTTCCCTTCTGACCTGGGTTTCTCTCAGAGCAAGTAGCCGCTGCCTGTGGATTTGAGGATTTCCGAG
CTGAAGCAGGGATCCTGAATTACTACCGCCTGGACTCCACACTGGGAATCCACGTAGACAGATCTGAGCTAGATCACTCC
AAACCCCTTGCTGTCTTTCAGCTTTGGACAGTCCGCCATCTTTCTCCTGGGTGGTCTTCAAAGGGATGAGGCCCCACGGC
CATGTTTATGCACAGTGGTGACATCATGATAATGTGCGGTTTCAGCCGCTCTTGAACACGCGAGTCCCTCGTGTCTTC
CAAATCCAGAAGGGGAAGGCCTGCCTCACTGCCTAGAGGCACCTCTCCCTGCTGTCTCCCGAGAGATTCAATGGTAGAG
CCTTGTTCTATGGAGGACTGGCAGGTGTGTGCCAGCTTGAAGACCGCTCGTGTTAACATGACTGTCCGACAGGTCTC
GGCCACAGACCAGAAATTTCCCTCTAGAACCATCGAGGATGAAAAAGAGACATCAGTACAAGGTTTCTGCCATCTGG
ATGACCAGAATAGCGAAGTAAACGGCCAGGATAAACCTCAGACCTTG
b22feorp@AAVA1015:~/Descargas$
```


PRACTICA 2: HOJA DE CALCULO

(Esta practica se encuentra en el progreso 1 pero incompleta)

El objetivo de esta practica se centra en aprender a utilizar las hojas de cálculos, en este caso Google Sheets, aprenderemos a utilizarlas con el supuesto de las notas de una asignatura.

Durante esta practica hemos llevado a cabo una serie de tareas.

Primero hemos calculado la nota final de los alumnos con una ponderación según los criterios de esta asignatura.

Hay que tener en cuenta que para auto rellenar una celda con una formula anterior hay que seleccionar la celda con la formula que queremos y “arrastrarla” y que para fijar un valor en la formula hay que poner \$ antes de cada valor que queremos fijar (una antes de la fila y otro antes de la columna)

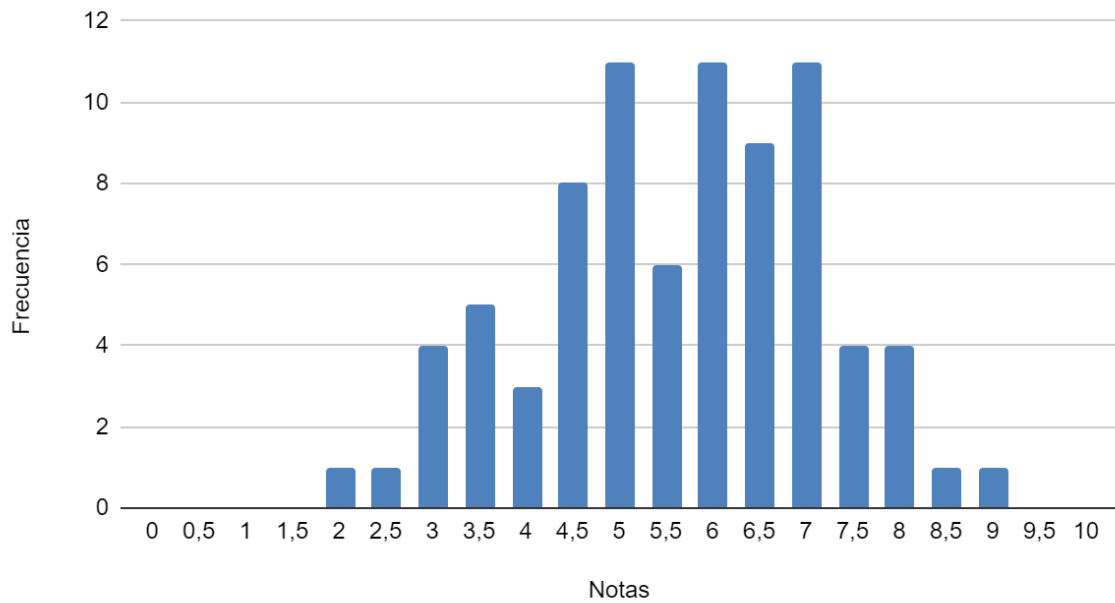
Hemos utilizado esta fórmula =SUMA(A2*0,2;B2*0,6;C2*0,2)

Ahora calcularemos la frecuencia de los datos y hemos hecho un histograma sobre ello. Para hallar la frecuencia hemos utilizado la siguiente función

- FRECUENCIA(datos; clases). Calcula la frecuencia de valores dentro de un conjunto de datos determinado.

Y para realizar el histograma debemos seleccionar tanto los datos de los que hemos sacado la frecuencia y esta última. Con los datos aportados hemos obtenido esta grafica.

Distribución de notas finales



Lo siguiente que vamos a llevar a cabo es la aplicación de un formato condicional a las celdas. Para ello seleccionamos las celdas a las que queremos aplicarles el formato y con click derecho nos saldrá un menú donde deberemos seleccionar formato condicional y ya aplicamos el formato deseado

Nota Final
5,42
4,28
4,86
5,2
6,56
8,26
6,72
4,94
7,6
4,1
5,54
5,7
4,82
4,86

6,1
6,54
6,7
8,72
6,84
2,32
5,12
5,2
3,42
5,68
6,18
5,64
3,1
2,98
1,52
3,68
4,36
6,42
6,4
6,5
3,22
6,68
5,1
7,54
5
6,24
3,68
5,98
5,98
7,1
7,26
5,06
6,08

4,5
7,72
4,34
6,74
4,82
4,76
5,58
7,3
4,26
4,5
5,78
6,08
3,78
2,96
4,8
2,64
5,8
6,58
4,18
6,74
7,52
5,64
6,86
5,8
4,88
4,56
3,2
4,7
6,12
6,92
3,48
7,18
2,88

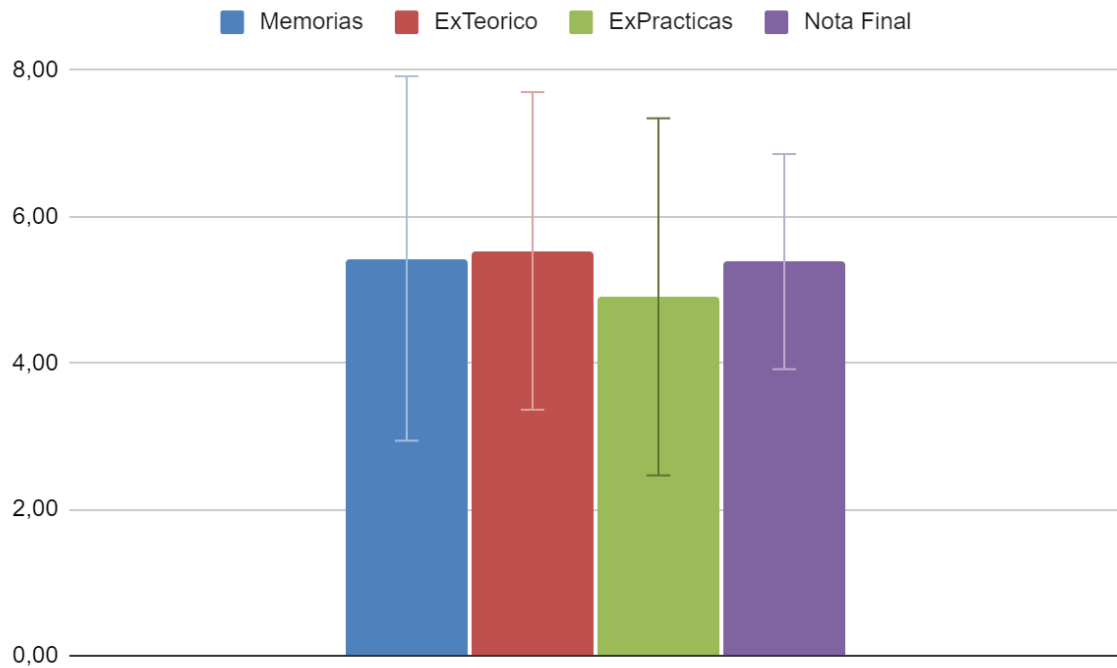
5,39

Ahora llevaremos a cabo una serie de cálculos estadísticos sobre estas notas. Utilizamos las siguientes formulas

- PROMEDIO(valor1; valor2). Nos calcula el valor promedio en una lista de valores
- STDEV(valor1; valor2). Nos calcula la desviación típica en una serie de datos.
- VAR(valor1, valor2). Calcula la varianza de una serie de datos
- NORMALIZACION(valor; media; desviación_estándar). Calcula el valor equivalente normalizado de una variable a partir de la media y la desviación estándar de la distribución

78	6,6	8	4	6,92
79	2,5	4	2,9	3,48
80	7,6	9,2	0,7	7,18
81	0	3,5	3,9	2,88
82	MEDIAS			
83	5,43	5,53	4,90	5,39
84	DESVIACIÓN TIPICA			
85	2,499548709	2,174947916	2,449869576	1,474484656
86	VARIANZA			
87	6,326829114	4,790276899	6,077833861	2,201625316
88				
89				

Con estos datos crearemos una grafica de columnas que nos muestre los valores medios y su desviación típica. En esta grafica añadiremos las barras de error que pondremos con la constante de su desviación típica.



Además, crearemos una nueva columna con las notas finales estandarizadas utilizando la función “Normalización”, donde “valor” será la nota final, “media” la media de la nota final calculada previamente con la función “promedio” y “desviación estándar”, la desviación típica previamente calculada con la función “Stdev”. En mi caso la función será:

= NORMALIZACION(D2; \$D\$82; \$D\$84)

0,02373710697
-0,7494143772
-0,3560566046
-0,1254675654
0,7968885912
1,949833787
0,9054010802
-0,3018003601
1,50221977
-0,8714909274
0,1051214737
0,2136339627
-0,3831847268

-0,3560566046
0,4849151853
0,78332453
0,891837019
2,261807193
0,9867854469
-2,078692368
-0,1797238099
-0,1254675654
-1,332669006
0,2000699016
0,5391714298
0,1729417794
-1,549693984
-1,63107835
-2,621254813
-1,156336211
-0,6951581327
0,7019401633
0,6883761022
0,7561964078
-1,468309617
0,8782729579
-0,193287871
1,461527586
-0,2611081767
0,5798636131
-1,156336211
0,4035308185
0,4035308185
1,163118242
1,271630731
-0,2204159933

0,4713511241
-0,6002097048
1,583604136
-0,7087221938
0,9189651413
-0,3831847268
-0,4238769102
0,132249596
1,298758853
-0,7629784383
-0,6002097048
0,2678902072
0,4713511241
-1,088515905
-1,644642412
-0,3967487879
-1,86166739
0,2814542684
0,8104526523
-0,8172346829
0,9189651413
1,447963525
0,1729417794
1,000349508
0,2814542684
-0,3424925434
-0,5595175215
-1,481873678
-0,4645690936
0,4984792464
1,041041691
-1,291976822
1,217374486

-1,698898656

Y por último como tarea para casa había que realizar una grafica de tipo pie chart con las calificaciones cualitativas. Para ello primero hay que sacar las calificaciones cualitativas.

Para ello utilizamos la siguiente función:

- SI(expresión_lógica, valor_si_verdadero, valor_si_falso). Muestra un valor si una expresión lógica es "VERDADERA" y otra si es "FALSA"

Para que en nuestra hoja de calculo aparezca suspenso, aprobado, notable o sobresaliente utilizamos la siguiente formula:

= SI(A2 < 5; "suspenso"; SI(A2 < 7; "aprobado"; SI(A2 < 9; "notable"; SI(A2 > 9; "sobresaliente"))))

1,1	suspenso
4,8	suspenso
7,3	notable
10	sobresaliente
6,3	aprobado
6	aprobado
7,6	notable
6,1	aprobado
8,2	notable
4,5	suspenso
5,3	aprobado
3,4	suspenso
3,1	suspenso
3,5	suspenso
7	notable
5,2	aprobado
0,4	suspenso
9,6	sobresaliente
6,6	aprobado

1,1	suspenso
4,1	suspenso
5,5	aprobado
3,1	suspenso
4,7	suspenso
5,8	aprobado
7,6	notable
1,5	suspenso
2,4	suspenso
3,4	suspenso
5,4	aprobado
8,4	notable
9,8	sobresaliente
6,9	aprobado
7	notable
4,6	suspenso
4,8	suspenso
3	suspenso
6,2	aprobado
8,1	notable
8,2	notable
4,3	suspenso
5,5	aprobado
7,7	notable
6,3	aprobado
10	sobresaliente
4,6	suspenso
6,1	aprobado
0,1	suspenso
3,6	suspenso
0,3	suspenso
4,7	suspenso
1,7	suspenso

3,3	suspenso
2,7	suspenso
4,6	suspenso
7,4	notable
8,9	notable
6,1	aprobado
3,6	suspenso
5,4	aprobado
3,1	suspenso
2,6	suspenso
1,5	suspenso
0,2	suspenso
4,8	suspenso
6,7	aprobado
5,5	aprobado
5	aprobado
5,1	aprobado
3,6	suspenso
9,7	sobresaliente
4,5	suspenso
1,5	suspenso
4,2	suspenso
3,6	suspenso
4,7	suspenso
4	suspenso
2,9	suspenso
0,7	suspenso
3,9	suspenso
4,90	suspenso

Ahora llevamos a cabo el gráfico tipo pie chart para eso debemos hacer un recuento de cuantos suspensos, aprobados, notables y sobresalientes hay. Para ello utilizamos la función:

- **CONTAR.SI(intervalo; criterio).** Cuanta las celdas que cumplan el criterio asignado.

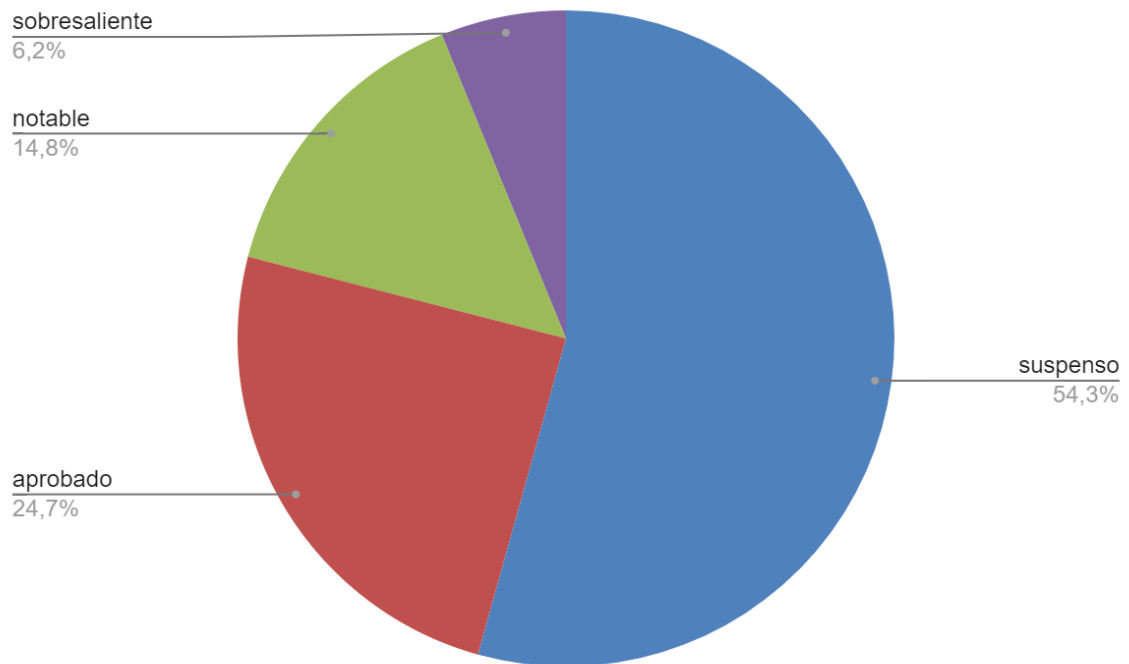
En nuestro caso la función para contar los suspensos será:

= CONTAR.SI(B2:B82; "suspense")

Utilizamos la misma función, pero cambiando el criterio para cada una de las opciones.

D	E
44	suspense
20	aprobado
12	notable
5	sobresaliente

Después seleccionamos los datos y creamos el diagrama pie chart.



Indica las diferencias/mejoras que aporta Excel en Office 365 vs Google Sheets.

Excel en Office 365 sobresale en análisis de datos, automatización avanzada y manejo de datos complejos. Se integra a la perfección con otras aplicaciones de Microsoft, lo que facilita la colaboración en tiempo real en entornos corporativos. Ofrece un mayor control de acceso y personalización, además de un soporte técnico de nivel empresarial. También se destaca por su compatibilidad con formatos de archivo de Excel.

En cambio, Google Sheets brilla por su acceso en línea desde cualquier dispositivo, colaboración en tiempo real, almacenamiento en la nube y facilidad para compartir y acceder a documentos. Además, ofrece una versión gratuita básica, funciona en diversas plataformas y se integra con otros servicios de Google. Una comunidad activa en línea proporciona recursos adicionales.

En resumen, Excel en Office 365 se enfoca en capacidades avanzadas y colaboración empresarial, mientras que Google Sheets destaca por su accesibilidad en línea y facilidad de colaboración. La elección depende de tus necesidades y preferencias.

PRACTICA 3. EDICIÓN DE TEXTOS CIENTIFICOS CON LATEX

El objetivo de esta práctica sirve para familiarizarte con el uso de LaTeX, es una herramienta para componer documentos de aspecto profesional. Nos encontramos ante un sistema que emplea comandos para presentar contenidos complejos como pueden ser libros o artículos científicos. En este caso utilizaremos TeXstudio.

Copiar el ejemplo modelo

Una cosa recomendable es escribir comentarios que no saldrán en nuestro documento utilizando el símbolo %

Lo primero que debemos hacer es copiar la plantilla que nos sea proporcionada en nuestro caso es

```
"\documentclass[a4paper,12pt]{article}  
\usepackage[utf8x]{inputenc}  
\usepackage[T1]{fontenc}  
\usepackage[spanish]{babel}  
\usepackage[pdftex]{graphicx}  
\usepackage{color}"
```

Siempre nos debemos asegurar de escribir `\begin{document}` al principio y `\end{document}` al final de nuestro documento.

Para crear un título con nuestro nombre y la fecha de entrega debemos escribir antes de `\begin{document}`:

```
\title{El título de nuestro documento}  
\author{el nombre del autor, en nuestro caso nuestro nombre}  
\date{la fecha de entrega en nuestro caso}
```

Y después de `\begin{document}` debemos añadir `\maketitle` para que aparezca.

Para crear las secciones, subsecciones y subsubsecciones debemos escribir respectivamente `\section`, `\subsection`, `\subsubsection`.

A la hora de darle formato al texto debemos utilizar `\textbf{texto que quieras en negrita}`, `\textit{texto que quieras en cursiva}` y `\underline{texto que quieras subrayado}`.

Para el uso de ecuaciones debemos hacerlo de la siguiente forma:

`\begin{equation}`

Aquí incluimos la ecuación que deseemos incluir

`\end{equation}`

```
\section{Uso de ecuaciones}
\begin{equation} \label{ecuación 1}
s=\pi r^2
\end{equation}
Uno puede referirse a ecuaciones así: \ref{ecuaci
```

$$s = \pi r^2$$

A veces necesitamos añadir variables del tipo α o γ ; esto se hace usando el símbolo `\$`.

Ahora explicaremos el uso de las tablas para empezar debemos iniciarla con `\begin{table}[tbhp]`, si la queremos centrar debemos utilizar `\begin{center}` (también aplicable al resto del documento), también podemos tabular con `\begin{tabular}` {debemos incluir como queremos tabular, incluyendo las líneas verticales que queremos que haya y poniendo una letra por valor}, en el ultimo caso l= izquierda, r=derecha y c = centro

Para poner las líneas horizontales de la tabla hay que utilizar `\hline`.

A la hora de colocar los datos debemos colocar `&` después de cada dato que queramos separar.

Una vez terminada nuestra tabla no se nos puede olvidar poner `\end{table}` y en caso de que lo hayamos centrado y tabulado `\end{center}` y `\end{tabular}` respectivamente.

```

\begin{table}[t]
\begin{center}
\begin{tabular}{|l|c|c|l|r|}
\hline
Fila 1 & 1 & 2 & 3 & total \\
\hline
Números & 4 & 5 & 6 & avg \\
\hline
\end{tabular}
\end{center}
\caption{esta tabla contiene números consecutivos.}
\label{tab 1}
\end{table}

```

Línea: 64 Columna: 11 INSERTAR

mensajes Registro Avance Resultados de la Búsqueda

ceso iniciado: pdflatex -synctex=1 -interaction=nonstopmode "documento modelo".tex

2. Uso de ecuaciones

$$s = \pi r^2 \quad (1)$$

Uno puede referirse a ecuaciones así: 1. También se pueden mencionar secciones de la misma forma: ver sección 1.1

A veces necesitamos añadir variables del tipo α o γ ; esto se hace usando el símbolo \$.

3. Uso de tablas

Fila 1	1	2	3	total
Números	4	5	6	avg

Cuadro 1: esta tabla contiene números consecutivos.

Para hacer referencias primero deberemos poner una etiqueta con `\label{nombre}` a lo que queramos hacer referencia y luego cuando lo queramos citar utilizar `\ref{nombre que hemos utilizado}`

`\label{tab 1}`

`\end{table}`

En la Tabla `\ref{tab 1}` mostramos un ejemplo de tabla en `\LaTeX`. `\\`

Con `\LaTeX` se escribe el logo de LaTeX y con `\\` se hace un salto de línea.

Para incluir imágenes debemos asegurarnos que esté `\usepackage[pdftex]{graphicx}` antes de `\begin{document}`. Ahora deberemos poner

`\begin{figure}[tbhp]`

`\includegraphics[Tamaño que quieras poner la imagen en mm]{nombre de la imagen}`

`\end{figure}`

documento modelo.tex

```

\hline
\hline
Números & 4 & 5 & 6 & avg \\
\hline
\end{tabular}
\end{center}
\caption{esta tabla contiene números consecutivos.}
\label{tab 1}
\end{table}

En la Tabla \ref{tab 1} mostramos un ejemplo de tabla en \LaTeX. \\

\begin{figure}[tbbp]
\includegraphics[width=140 mm]{tux.png}
\caption{En esta figura se muestra a la mascota de Linux con 140mm de ancho.}
\label{imagen 1}
\end{figure}

\section{Uso de figuras}

La figura \ref{imagen 1} muestra varias versiones de la mascota de Linux: Tux.\\

\begin{figure}[tbbp]
\centerline{\includegraphics[width=0.5 \textwidth]{tux.png}}
\caption{En esta figura se muestra en modelo atómico de Rutherford con ancho igual a la mitad del ancho del texto y centrado}

\label{imagen 2}
\end{figure}

La figura \ref{imagen 2} también muestra a Tux, pero a un tamaño menor.

\end{document}

```

Línea: 65 Columna: 126 INSERTAR

nsajes Registro Avance Resultados de la Búsqueda

ceso iniciado: pdflatex -synctex=1 -interaction=nonstopmode "documento modelo".tex

roceso terminó normalmente

En la Tabla 1 mostramos un ejemplo de tabla en \LaTeX.




Figura 1: En esta figura se muestra a la mascota de Linux con 140mm de ancho.

Debes asegurarte que la imagen este guardada en la misma carpeta que tu documento.

Para poner el tamaño de la foto en relación con el tamaño pones [width=(relación) |textwidth] en vez del tamaño en milímetros, por ejemplo [width=0.5 \textwidth]

documento modelo.tex

```

\hline
\hline
Números & 4 & 5 & 6 & avg \\
\hline
\end{tabular}
\end{center}
\caption{esta tabla contiene números consecutivos.}
\label{tab 1}
\end{table}

En la Tabla \ref{tab 1} mostramos un ejemplo de tabla en \LaTeX. \\

\begin{figure}[tbbp]
\includegraphics[width=140 mm]{tux.png}
\caption{En esta figura se muestra a la mascota de Linux con 140mm de ancho.}
\label{imagen 1}
\end{figure}

\section{Uso de figuras}

La figura \ref{imagen 1} muestra varias versiones de la mascota de Linux: Tux.\\

\begin{figure}[tbbp]
\centerline{\includegraphics[width=0.5 \textwidth]{tux.png}}
\caption{En esta figura se muestra en modelo atómico de Rutherford con ancho igual a la mitad del ancho del texto y centrado}

\label{imagen 2}
\end{figure}

La figura \ref{imagen 2} también muestra a Tux, pero a un tamaño menor.

\end{document}

```

Línea: 65 Columna: 126 INSERTAR

nsajes Registro Avance Resultados de la Búsqueda

oceso iniciado: pdflatex -synctex=1 -interaction=nonstopmode "documento modelo".tex

proceso terminó normalmente

En esta figura se muestra en modelo atómico de Rutherford con ancho igual a la mitad del ancho del texto y centrado




Figura 2: En esta figura se muestra en modelo atómico de Rutherford con ancho igual a la mitad del ancho del texto y centrado

PRACTICA 4. BASES DE DATOS BIBLIOGRÁFICA

En esta práctica el objetivo es aprender a utilizar gestores bibliográficos, en especial Endnote, Mendeley y Google Scholar.

Los gestores bibliográficos son herramientas que se utilizan para organizar la bibliografía consultada durante un trabajo escrito.

WUOLAH

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

ENDNOTE.

Para empezar, utilizaremos la plataforma online de Endote, antes de nada, debemos crear nuestra cuenta utilizando nuestro correo UCO y activando esta cuenta.

Para crear un grupo donde guardar nuestra bibliografía accedemos a Organice y seleccionamos Manage my groups. Crearemos un grupo llamado PracticasBioq.

Para añadir referencias a nuestro grupo le damos a Collect y seleccionamos Online search, en nuestro caso seleccionamos PubMed MEDLINE para buscar la información. Para hacer una búsqueda tendremos que seleccionar la información que queramos buscar en campos concretos, en este caso utilizaremos Universidad de Córdoba en el campo Autor.

Online Search

Step 2 of 3: Connecting to PubMed MEDLINE : PubMed (NLM)

Universidad de Córdoba	in:	Author (Smith AB) ▾	And ▾
	in:	Any Field ▾	And ▾
	in:	Any Field ▾	And ▾
	in:	Any Field ▾	

- ☒ retrieve all records
☐ select a range of records to retrieve

Search

Seleccionamos los artículos que deseamos guardar y le damos a add to group (seleccionamos el nombre del grupo al que lo queramos añadir).

Online Search Results

PubMed MEDLINE 1 - 10 of 18 results
(Universidad de Córdoba) in Author (Smith AB)

Show 10 per page

Page 1 of 2 Go

Author	Year	Copied Title
<input type="checkbox"/> Marcelino, A. A.	2022	- New reference values for maximum respiratory pressures in healthy Brazilian children following g PLoS One Online Link→ Go to URL Full Text
<input type="checkbox"/> de-Córdoba, G. F.	2022	- Sankey diagrams for macroeconomics: A teaching complement bridging undergraduate and grad Heliyon Online Link→ Go to URL
<input type="checkbox"/> Millán-Sánchez, V.	2022	- Reply to: Role of ambient humidity underestimated in research on correlation between radioactiv Sci Rep Online Link→ Go to URL Full Text
<input type="checkbox"/> Leite, H. R.	2021	- Measurement properties of the six-minute pegboard and ring test (6PBRT) in healthy adolescents J Bodyw Mov Ther Online Link→ Go to URL
<input type="checkbox"/> Stanzani, V.	2018	- Performance and Reproducibility of the Continuous Scale Physical Functional Performance 10 Test CROSS-SECTIONAL STUDY J Cardiopulm Rehabil Prev Online Link→ Go to URL Full Text
<input type="checkbox"/> Klionsky, D. J.	2016	- Guidelines for the use and interpretation of assays for monitoring autophagy (3rd edition) Autophagy Online Link→ Go to URL Full Text
<input type="checkbox"/> Rabasco, C.	2015	- Effectiveness of mycophenolate mofetil in C3 glomerulonephritis Kidney Int Online Link→ Go to URL Full Text
<input type="checkbox"/> Triana, J.	2014	- Generation and Evaluation of a Genome-Scale Metabolic Network Model of Synechococcus elonga Metabolites Online Link→ Go to URL Full Text
<input type="checkbox"/> Gamermann, D.	2014	- New approach for phylogenetic tree recovery based on genome-scale metabolic networks

Ahora para exportar estas referencias y poder citarlas en un fichero Latex debemos darle a format y seleccionar export references, en export style usaremos BibTex. Es importante que guardemos el fichero en formato .bib. En nuestro caso lo guardaremos como Endnote.bib.

Ahora para poder citar en un documento Latex debemos abrir un documento .tex en textstudio y antes de `\end{document}` escribimos en nuestro caso:

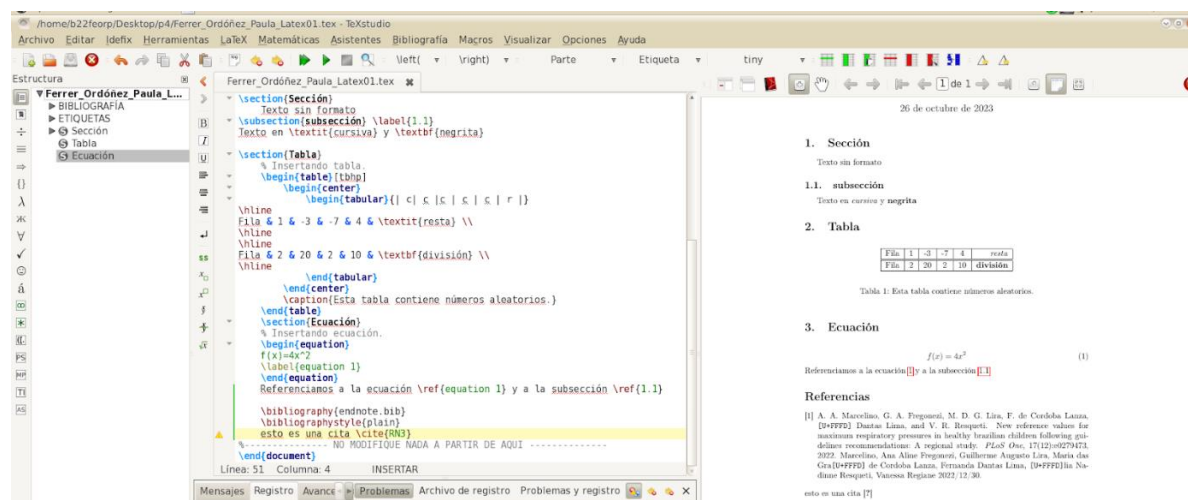
```
\bibliography{endnote.bib}
```

```
\bibliographystyle{plain}
```

```
esto es una cita \cite{RN3}
```

Deberemos sustituir endnote.bib por cualquier fichero que contenga las referencias que queremos añadir y en RN3 la cita que queramos según lo que ponga el fichero bibliográfico.

```
endnote.bib x
@article{RN3,
  author = {de-Córdoba, G. F. and Molinari, B.},
  title = {Sankey diagrams for macroeconomics: A teaching complement bridging undergraduate and graduate Macro},
  journal = {Heliyon},
  volume = {8},
  number = {9},
  pages = {e10717},
  note = {de-Córdoba, Gonzalo F
Molinari, Benedetto
2022/10/5},
```

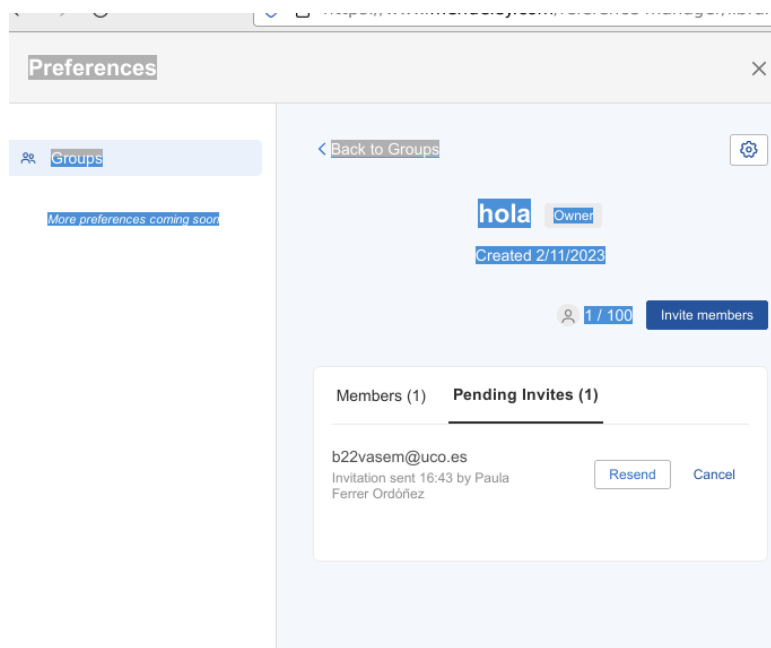


MENDELEY

Ahora utilizaremos la plataforma online Mendley. Para ello debemos crear una cuenta utilizando el correo UCO.

Para añadir referencias a una colección lo debemos hacer desde library.

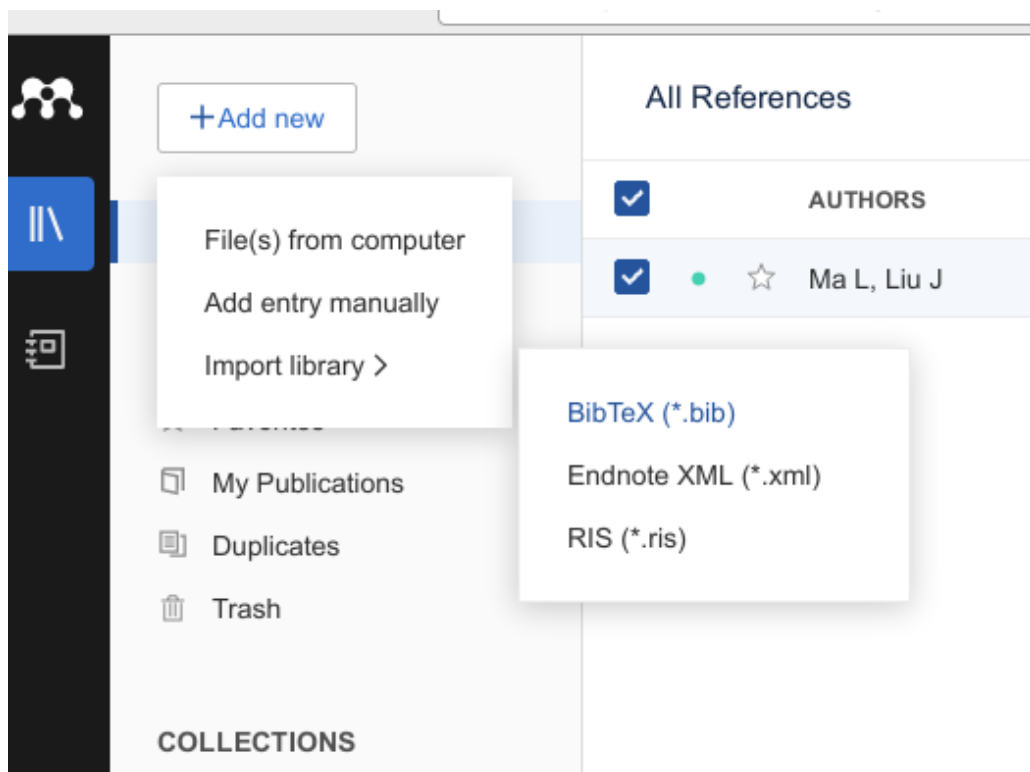
Ahora crearemos un grupo, en mi caso se llama hola, para añadir a un compañero debemos seleccionar el grupo y darle a los tres puntos y seleccionar manage group y escribimos el correo de nuestro compañero.



Para añadir un grupo público debemos realizar la búsqueda con nuestro tema de interés en la página principal, en este caso biochemistry. Debemos darle a Add to library.

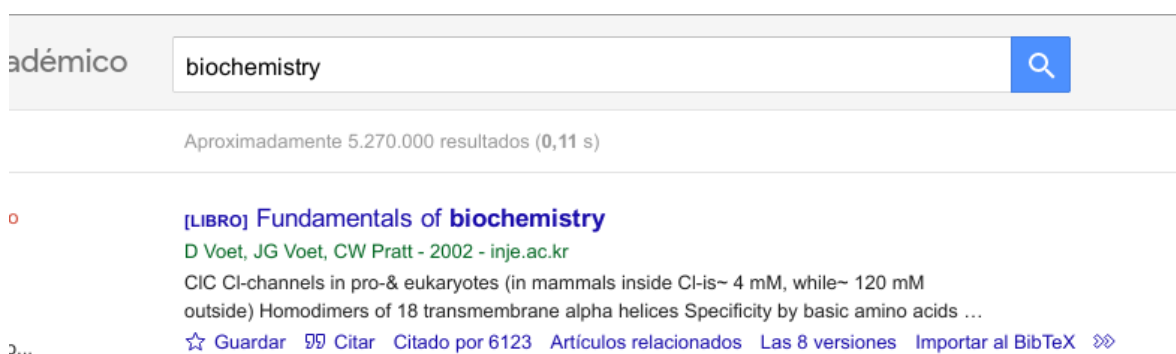
Para exportar esta referencia en formato BibTex debemos irnos a library y seleccionamos lo que queremos exportar.

También podemos importar nuestras propias referencias utilizando la opción import library y seleccionamos el formato que deseemos.



GOOGLE SCHOLAR

Por ultimo exportaremos nuestras referencias desde Google Scholar para ello primero habilitar la opción Mostrar enlaces para importar citas a Bibtex en la configuración y ahora cuando relicemos una busqueda nos saldrá la opción de importar a BibTeX.



```
@book{voet2002fundamentals,
  title={Fundamentals of biochemistry},
  author={Voet, Donald and Voet, Judith G and Pratt, Charlotte W and others},
  number={QD415 V63},
  year={2002},
```

publisher={Wiley New York}
}

PRACTICA 5 : INTRODUCCIÓN A R STUDIO

En esta práctica el objetivo es familiarizarse con el lenguaje R y hacer algoritmos que se adecuen a las peticiones.

Algunos comandos y símbolos que utilizaremos en los ejercicios de las diferentes partes a continuación son:

Los operadores básicos son **+, -, /, *, ^**.

Exp. Nos da el exponente **n** del número **e**

#. Se utiliza para hacer comentarios

Nombre de la variable <- scan(n=x). Para que el usuario introduzca **x** número de variables bajo un nombre, por ejemplo, si **n<-scan(n=1)**, la variable **n** pasará a valer el valor que introduzca el usuario hasta que se vuelva a cambiar, si solo ponemos **scan()** se podrán introducir todos los valores que queramos hasta pulsar intro, si en vez de poner **scan(n=X)** ponemos **n<-1**, **n** valdrá 1.

Print("Texto que queremos mostrar"). Lo utilizaremos para mostrar solo cadenas de texto sin ninguna variable.

{} utilizaremos estas llaves para mostrar de donde hasta dónde van los comandos.

If(condición){} Se utiliza para realizar un comando si una condición es verdadera

Else{} Se utiliza para llevar a cabo un comando si la condición de **if** no es verdadera, no es obligatorio utilizarlos juntos.

Stop("mensaje de error"). Para el algoritmo dando un mensaje de error.

Cat("mensaje que queremos mostrar"). Mostrará un mensaje pero en este caso también podrá añadir variables por ejemplo **cat("n vale", n)**, las variables deberán estar separadas por una coma.

While(condición){}. Repite un comando hasta que la condición no se cumpla.

==. Idéntico

!=. Distinto

<=. Menor o igual

%/%. Te devolverá la división sin decimales

%% Te devolverá el resto de la división

||. Significa o

Estructura switch case. Nos permite seleccionar valores de una lista, para ello primero deberemos de seleccionar una opción con **x<-scan(n=1)**, luego con el comando:


```
switch(x,
      "a" = "uno"
      "b" = "dos"
      ...
      Error)
```

De este modo al introducir una de las opciones nos mostrará el correspondiente valor, sin embargo si introducimos un valor que no esté dentro de las opciones nos devolverá el valor que no tenga asignado ninguna opción, si lo que introducimos son valores numéricos sería de la siguiente forma:

```
Switch(x,
      Uno
      Dos
      ...)
```

Al introducir el número nos devolverá el valor que se encuentre en esa posición

For (variable in a:b). Esta función sirve para llevar a cabo una función un determinado numero de veces, la variable será el "contador" de las veces que llevemos a cabo la función y cambiará de valor en cada ciclo de la función, a:b, hace referencia desde que numero hasta que numero irá la variable, por ejemplo en for (i in 1:10) la i cambiará de valor en las operaciones del 1 hasta el 10.

Source. Sirve para abrir un archivo R guardado en el directorio de trabajo.

sample(desde que número: hasta que numero, cantidad de números que queremos generar). Nos generará un numero aleatorio siguiendo los valores establecidos en la función.

Para añadir un **vector** en Rstudio deberemos seguir la siguiente forma `nombre_de_la_variable <- c(números dentro del vector separados por comas)`, si queremos un vector vacío solo pondremos `c()`, también podemos crearlo con una serie de números seguidos con `c(1:9)`, en este caso el vector tendrá los números del 1 al 9, también podemos unir dos vectores con `c(v1,v2)` y podemos crear una secuencia de números no seguidos con **seq(desde que numero, hasta que numero, intervalo entre números)**

Sd(x). Calculará la desviación típica de un vector x.

sum(v1). Calculará la suma de todos los números del vector v1.

length(v1). Nos dirá el número de datos que tiene nuestro vector.

Min(v1). Nos mostrará el valor más pequeño del vector

Max(v1). Nos mostrará el valor más grande del vector.

Para acceder a los elementos de un vector utilizamos **v1[i]**, donde i será la posición que queramos ver, si queremos eliminar un valor del vector **v1[-i]**, donde i será la posición que queremos eliminar.

Para que crear una matriz hay varias opciones, podemos poner **matrix(vector, numero filas, numero columnas)**, si queremos que el vector se coloque por columnas o **matrix(vector, numero filas, numero columnas, TRUE)**, si queremos que el vector se coloque por filas.

Para acceder a los valores de una matriz utilizaremos **M1[i,k]**, donde i será el número de la fila y k, el número de la columna. Si solo queremos ver los valores de una fila dejaremos la parte de la columna vacía.

Para unir dos matrices hay dos opciones **cbind(M1,M2)**, que las unirá desde las columnas, y **rbind(M1,M2)**, que las unirá por las filas.

Dim(M1). Nos dirá las dimensiones de nuestra matriz, primero las filas y luego las columnas.

Nrow(M1). Numero de filas

Ncol(M1). Numero de columnas

Para sumar matrices simplemente utilizaremos **M1+M2**, pero para multiplicarlas **%*%** ya que si solo utilizamos ***** multiplicará elemento a elemento.

Para otras operaciones con matrices será necesario cargar **library(MASS)**

T(M). Nos da la matriz transpuesta.

Ginv(M). Nos da la matriz inversa.

Det(M). Nos da el determinante.

Además, utilizaremos la **R notebook**:

En RStudio, puedes crear cuadernos de R Markdown que combinan código R, resultados de código, y texto formateado. Esto proporciona una forma efectiva de realizar análisis de datos, presentar resultados y documentar tu trabajo, todo en un solo documento, para añadir un código R a tu libreta R deberás añadir antes `{r}` y volver a cerrar con ````` a esto se le conoce como chunk.

Ejemplo:

```
```{r}

print("Este es mi primer cuaderno")
```

```
x <- 16 / 4 * 4
cat("El resultado de la operacion es", x, "\n")
````
```

También podrás añadir funciones como en latex utilizando \$\$ antes y después de la formula.

Ejemplo:

```
$$f(x) = \frac{1}{\sigma^2 \cdot \sqrt{2}}$$
```

También podremos añadir texto en negrita y en cursiva, añadiremos ******, antes y después del texto, cuando queramos *negrita* y *** cuando lo queramos en cursiva.

Si en un chunk no queremos que salgan las operaciones llevadas a cabo pondremos:

```
```{r echo=FALSE}
y <- 16 / (4 * 2)
print(y)
```
```

Y lo único que veremos en nuestra notebook una vez le hayamos dado a “Run All” será el valor de y.

Para abrir un código que tenemos guardado en nuestro directorio de trabajo utilizaremos:

```
```{r setup}
knitr::read_chunk("test00.R")
```
```

Y una vez le hayamos dado a run saldrá el código R que tenemos en nuestro archivo. Y con:

```
```{r chunk}
```
```

Nos mostrará el resultado. Para que poder abrir el archivo debemos poner **## ----** chunk en el script R que queramos añadir.

Para poner texto de un tamaño mayor utilizaremos # antes de este, es importante dejar el espacio.

Ejercicios parte 1.

Ejercicio A.

```
# -----  
#' Archivo: p5_parte1_eja.R  
#' @description Convierte euros en pesetas  
#' @author Paula Ferrer Ordoñez  
#' Fecha: 9 noviembre 2023  
# -----  
  
#Introducimos los euros que queremos pasar a pesetas  
  
print("Introduzca la cantidad de euros:")  
    euros<- scan(n=1)  
#si el número de euros es inferior a 0 no tiene sentido llevar a cabo la conversion  
  
    if(euros < 0) {  
        stop("La cantidad de euros debe ser superior a 0");  
    }  
  
#rerealizamos la conversiom  
  
ptas<-euros*166.386  
  
cat(euros, "euros son", ptas," pesetas")
```

Ejercicio B.

```
# -----  
#' Archivo: p5_parte1_ejb.R
```

```
#' @description Calcular el área de un triángulo conocida su base y su altura.
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 9 noviembre 2023
```

```
# -----
```

```
#Debemos pedir que se Introduzca los valores de la base y la altura
```

```
print("Introduzca la base")
```

```
base<- scan(n=1)
```

```
#Hay que tener en cuenta que la base no puede ser ni negativa ni 0
```

```
if(base<=0){
```

```
  stop("La base no puede ser ni 0 ni negativa")
```

```
}
```

```
print("Introduzca la altura")
```

```
altura<- scan(n=1)
```

```
#Hay que tener en cuenta que la altura no puede ser ni negativa ni 0
```

```
if(altura<=0){
```

```
  stop("La altura no puede ser ni 0 ni negativa")
```

```
}
```

```
#teniendo en cuenta que la fomula del area de un triangulo es (base*altura)/2 realizamos los calculos
```

```
area<- (base*altura)/2
```

```
cat("El area del triangulo sera", area)
```

Ejercicio C.

```
# -----  
#' Archivo: p5_parte1_ejc.R  
#' @description Calcular el área de un triángulo conocidos sus tres lados.  
#' @author Paula Ferrer Ordoñez  
#' Fecha: 9 noviembre 2023  
# -----  
  
#Pedimos los valores de los tres lados  
  
print("introduzca el valor del lado mayor")  
lado1<- scan(n=1)  
  
#Hay que tener en cuenta que los lados no puede ser ni negativos ni 0  
  
if(lado1<=0){  
  stop("ningún lado puede valer 0 o menos")  
}  
print("introduzca el valor del lado 2")  
lado2<- scan(n=1)  
  
#Hay que tener en cuenta que los lados no puede ser ni negativos ni 0  
  
if(lado2<=0){  
  stop("ningún lado puede valer 0 o menos")  
}  
print("introduzca el valor del lado 3")  
lado3<- scan(n=1)
```

```
#Hay que tener en cuenta que los lados no puede ser ni negativos ni 0
```

```
if(lado3<=0){  
  stop("ningún lado puede valer 0 o menos")  
}
```

```
if(lado2+lado3<lado1){  
  stop("la sumo de los lados mas pequeños debe ser mayor que el lado mayor")  
}
```

```
#realizamos los calculos
```

```
area<- (lado1 + lado2 + lado3)/2.
```

```
#mostramos los resultados
```

```
cat("El area del triangulo es", area)
```

EJERCICIO D.

```
# -----
```

```
#' Archivo: p5_parte1_ejd.R
```

```
#' @description Calcular la hipotenusa de un triángulo rectángulo conocidos sus dos catetos.
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 9 noviembre 2023
```

```
# -----
```

```
#Pedimos la información de los dos catetos
```

```
print("introduzca el valor del cateto 1")
```



```
cat1<- scan(n=1)
```

```
#Hay que tener en cuenta que los catetos no puede ser ni negativos ni 0
```

```
if(cat1<=0){  
  stop("ningún cateto puede valer 0 o menos")  
}
```

```
print("introduzca el valor del cateto 2")
```

```
cat2<- scan(n=1)
```

```
#Hay que tener en cuenta que los catetos no puede ser ni negativos ni 0
```

```
if(cat2<=0){  
  stop("ningún cateto puede valer 0 o menos")  
}
```

```
#Realizamos los calculos
```

```
hip<-sqrt((cat1*cat1)+(cat2*cat2))
```

```
#Mostramos los resultados
```

```
cat("La hipotenusa del triangulo es", hip)
```

EJERCICIO E.

```
# -----
```

```
#' Archivo: p5_parte1_eje.R
```

```
#'                                                                 @description  
Calcular el área de un trapezio conocidas la base mayor, menor y la altura:  
(areaT=[(bMa+bMe)·h]/2 )
```




```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 9 noviembre 2023
```

```
# -----
```

```
#Pedimos los datos
```

```
print("introduzca el valor de la base mayor")
```

```
bma<- scan(n=1)
```

```
#Hay que tener en cuenta que las bases no pueden ser ni negativos ni 0
```

```
if(bma<=0){
```

```
  stop("Las bases no pueden ser ni negativas ni 0")
```

```
}
```

```
print("introduzca el valor de la base menor")
```

```
bme<- scan(n=1)
```

```
#Hay que tener en cuenta que las bases no pueden ser ni negativos ni 0
```

```
if(bme<=0){
```

```
  stop("Las bases no pueden ser ni negativas ni 0")
```

```
}
```

```
print("introduzca el valor de la altura")
```

```
h<- scan(n=1)
```

```
#Hay que tener en cuenta que la altura no puede ser ni negativos ni 0
```

```
if(h<=0){
```

```
  stop("La altura no puede ser ni negativa ni 0")
```

```
}
```



```
#Realizamos los calculos
```

```
area<-(((bma+bme)*h) /2 )
```

```
#Mostramos los resultados
```

```
cat("El area del trapecio es", area)
```

EJERCICIO F.

```
# -----
```

```
#' Archivo: p5_parte1_ejF.R
```

```
#' @description Calcular el perímetro y el área de un círculo.
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 9 noviembre 2023
```

```
# -----
```

```
#Pedimos el radio
```

```
print("introduzca el valor del radio")
```

```
r<- scan(n=1)
```

```
#Hay que tener en cuenta que el radio no puede ser ni negativo ni 0
```

```
if(r<=0){
```

```
  stop("El radio no puede ser ni negativo ni 0")
```

```
}
```

```
#hacemos los calculos
```

```
per<- 2*pi*r
```

```
area<- pi*r^2
```

```
#Mostramos los resultados
```

```
cat("el perimetro es", per, "y el area es", area)
```

```
EJERCICIO G.
```

```
# -----
```

```
#' Archivo: p5_parte1_ejg.R
```

```
#' @description Calcular el logaritmo en base 2 de un número haciendo uso de la función log10  
de R. Pista: use el cambio de base
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 16 de noviembre de 2023
```

```
# -----
```

```
# Introducimos el número que queremos calcular
```

```
print("Introduzca el número del cual quiere calcular el logaritmo en base dos: ")
```

```
n<- scan(n=1)
```

```
# Comprobamos que es un número valido
```

```
if(n<=0){
```

```
  stop("No se puede el logaritmo de un numero negativo ni de 0")
```

```
}
```

```
# Realizamos el calculo
```

```
log<- log10(n)/log10(2)
```

```
# Mostramos el resultado obtenido
```

```
cat("el logaritmo en base 2 de", n, "es", log)
```

EJERCICIO H.

```
# -----
```

```
#' Archivo: p5_parte1_ejh.R
```

```
#' @description Calcular las raíces de un polinomio de grado 2:
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 16 de noviembre de 2023
```

```
# -----
```

```
#Pedimos los datos
```

```
print("Siguiendo la forma de un polinomio de segundo grado ( $p(x) = ax^2+bx+c$ )")
```

```
print("introduzca a")
```

```
a<- scan(n=1)
```

```
if(a==0){
```

```
  stop("Si no tiene valor a seria un polinomio de segundo grado")
```

```
}
```

```
print("introduzca b")
```

```
b <- scan(n=1)
```

```
print("introduzca c")
```

```
c <- scan(n=1)
```

```
#Hacemos los calculos
```

```
e <- (b^2)-(4*a*c)
```

```
if (e<0){
```

```

    print("No hay raices reales")
}
if (e>0){
  raiz <- sqrt(e)
  raiz1 <- (-b + raiz)/(2*a)
  raiz2 <- (-b - raiz)/(2*a)

  #Mostramos los resultados

  if (raiz1 == raiz2){
    cat("la raiz real es", raiz1)
  }
  else {
    cat("Las raices reales del polinomio son", raiz1, "y", raiz2)
  }
}

```

EJERCICIO I.

```

# ----
#' Archivo: p5_parte1_eji.R
#' @description Convertir una temperatura en grados Celsius (centígrados) en grados Fahrenheit
#' @author Paula Ferrer Ordoñez
#' Fecha: 16 de noviembre de 2023
# -----

#Pedimos los datos

print("Introduzca los grados en celsius que quieras pasar a Fahrenheit")
c<- scan(n=1)

```

```
#Llevamos a cabo los calculos
```

```
f <- (9/5)*c + 32
```

```
#Mostramos los resultados
```

```
cat(c, "celsius son", f, "fahrenheit")
```

EJERCICIO J.

```
# ----
```

```
#' Archivo: p5_parte1_ejj.R
```

```
#' @description Desglosar una cantidad de segundos en su equivalente en días, horas y minutos.
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 16 de noviembre de 2023
```

```
# -----
```

```
#Pedimos los datos
```

```
print("Introduzca los segundos que quieres desglosar")
```

```
s<- scan(n=1)
```

```
#Hacemos los calculos
```

```
m<- s/60
```

```
h<- m/60
```

```
d<- h/24
```

```
dias<- s/% (24*60*60)
```

```
sobras1<- s%(24*60*60)
```

```
horas<- sobras1%/%(60*60)
sobras2<- sobras1%%(60*60)
```

```
minutos<- sobras2%/60
segundos<- sobras2%60
```

```
#Mostramos los resultados
```

```
cat(s, "segundos seran", m, "minutos", h, "horas y",d, "dias o", dias, "dias", horas,
"horas",minutos, "minutos y", segundos, "segundos" )
```

EJERCICIO K

```
# ----
```

```
#' Archivo: p5_parte1_ejk.R
```

```
#' @description Calcular la nota final de un alumno en una asignatura
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 16 de noviembre de 2023
```

```
# -----
```

```
#Pedimos los datos
```

```
print("Introduzca su nota en el cuaderno de prácticas")
```

```
p <- scan(n=1)
```

```
if(p<0){
```

```
stop("La nota no puede ser negativa")
```

```
}
```

```
print("Introduzca su nota en el examen practico")
```

```
ep<- scan(n=1)
```

```

if(ep<0){
  stop("La nota no puede ser negativa")
}

print("Introduzca su nota en el examen teorico")

et<- scan(n=1)

if(et<0){
  stop("La nota no puede ser negativa")
}

#Hacemos los calculos

nf<- p*0.25 + ep*0.15 + et*0.6

#Mostramos los resultados

cat("Tu nota final sera", nf)

```

Ejercicios parte 2.

En los ejercicios 1 y 2 nos piden que utilicemos el comando stop para cuando demos al algoritmo un valor no valido y que a la hora de mostrar los resultados sea algo más informativo usando el comando cat, estos dos ejercicios se han hecho junto a la parte 1.

Ejercicio 3.

```

print("Programas disponibles:");
print("=====");
print("a - Convertir euros a pesetas.");

```



```

print("b - area del triangulo dada base y altura.");
print("c - Calcular el área de un triángulo conocidos sus tres lados.")
print("d - Calcular la hipotenusa de un triángulo rectángulo conocidos sus dos catetos.")
print("e - Calcular el área de un trapezio conocidas la base mayor, menor y la altura")
print("f - Calcular el perímetro y el área de un círculo")
print("g - Calcular el logaritmo en base 2 de un número haciendo uso de la función log10 de R.")
print("h - Calcular las raíces de un polinomio de grado 2")
print("i - Convertir una temperatura en grados Celsius (centígrados) en grados Fahrenheit ")
print("j - Desglosar una cantidad de segundos en su equivalente en días, horas y minutos.")
print("k - Calcular la nota final de un alumno en una asignatura")

```

```

opt <- readline(prompt="Elija una de las anteriores opciones: ");

```

```

switch(opt,
  a=source("p5_parte1_eja.R"),
  b=source("p5_parte1_ejb.R"),
  c=source("p5_parte1_ejc.R"),
  d=source("p5_parte1_ejd.R"),
  e=source("p5_parte1_eje.R"),
  f=source("p5_parte1_ejf.R"),
  g=source("p5_parte1_ejg.R"),
  h=source("p5_parte1_ejh.R"),
  i=source("p5_parte1_eji.R"),
  j=source("p5_parte1_ejj.R"),
  k=source("p5_parte1_ejk.R"),
  stop("Opcion no valida"))

```

EJERCICIO 4.

A.

```
#' Archivo: p5_parte2_ej4a.R
```

```
#' @description Un programa que lea un número entero e indique si es mayor, menor, o igual que cero
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 23 de noviembre de 2023
```

```
# -----
```

```
#Pedimos el numero
```

```
print("Introduzca el número entero:")
```

```
n<-scan(n=1)
```

```
if()
```

```
if(n==0){
```

```
  cat(n, "es igual que cero")
```

```
}
```

```
if(n<0){
```

```
  cat(n,"es menor que cero")
```

```
}
```

```
if(n>0){
```

```
  cat(n, "es mayor que cero")
```

```
}
```

```
B.
```

```
# ----
```

```
#' Archivo: p5_parte2_ej4b.R
```

```
#' @description raiz de un número
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 23 de noviembre de 2023
```

```
# -----
```

```
#Pedimos el número
```

```
print("Introduzca su numero entero")
```

```
n<- scan(n=1)
```

```
if(n<0){
```

```
  stop("el numero debe ser positivo")
```

```
}
```

```
r<- sqrt(n)
```

```
cat("la raiz de", n, "es", r)
```

```
C.
```

```
# ---- -----
```

```
#' Archivo: p5_parte2_ej4c.R
```

```
#' @description dias de dias en un mes
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 23 de noviembre de 2023
```

```
# -----
```

```
#Pedimos el mes
```

```
print("introduzca el numero del mes que quieres saber los dias")
```

```
n<- scan(n=1)
```

```
if(n<1 || n>12){
```

```
  stop("Los meses van del 1 al 12")
```

```

}

if(n==2){
  print("El mes tiene 28 o 29 dias")
}

if(n==1 || n==3 || n==5 || n==7 || n==8 || n==10 || n==12){
  print("El mes tiene 31 dias")
}

if(n==4 || n==6 || n==9 || n==11){
  print("el mes tiene 30 dias")
}

```

D.

```

# ----
#' Archivo: p5_parte2_ej4d.R
#' @description Dias de la semana
#' @author Paula Ferrer Ordoñez
#' Fecha: 23 de noviembre de 2023
# -----

```

```
#Pedimos los datos
```

```
dia <- readline(prompt="Escriba el numero del dia de la semana: ");
```

```

switch(dia,
  "1"= print("Lunes"),
  "2"=print("Martes"),
  "3"=print("Miercoles"),
  "4"=print("Jueves"),
  "5"=print("Viernes"),
  "6"=print("Sabado"),

```

```
"7"=print("Domingo"),  
stop("Los dias de la semana van del 1 al 7"))
```

Ejercicio 5.

```
---  
title: "Raizcuadrada"  
author: "Paula Ferrer Ordóñez GMA"  
date: "Noviembre 2023"  
output:  
  html_notebook: default  
  html_document: default  
  pdf_document: default  
---
```

Ejercicio 4b

Un programa que pida un número real al usuario por teclado e imprima la raízcuadrada del mismo. Como la raíz solo está definida para números positivos, si el número introducido es negativo, el programa no calcula la raíz e imprime por pantalla un mensaje de error

Utilizamos la estructura condicional `*if*` para evitar que se introduzcan numeros negativos

Llevaremos a cabo la siguiente funcion

$$f(x)=\sqrt{x}$$

Solucion

```
``{r}
```

```
print("Introduzca su numero entero")
```

```
n<- scan(n=1)
```

```
if(n<0){
```

```
stop("el numero debe ser positivo")
}
```

```
r<- sqrt(n)
```

```
cat("la raiz de", n, "es", r)
```

```
...
```

Raizcuadrada
Paula Ferrer Ordóñez GMA
Noviembre 2023

Ejercicio 4b

Un programa que pida un número real al usuario por teclado e imprima la raízcuadrada del mismo. Como la raíz solo está definida para números positivos, si el número introducido es negativo, el programa no calcula la raíz e imprime por pantalla un mensaje de error.

Utilizamos la estructura condicional *if* para evitar que se introduzcan números negativos.

Llevaremos a cabo la siguiente función

$$f(x) = \sqrt{x}$$

Solucion

```
print("Introduzca su numero entero")
n<- scan(n=1)

if(n<0){
  stop("el numero debe ser positivo")
}

r<- sqrt(n)

cat("la raiz de", n, "es", r)
```

Ejercicios Parte 3

Ejercicio A.

```
# -----
```

```
#' Archivo: p5_parte3_eja.R
```

```
#' @description Implementar un algoritmo que calcule la suma de los N primeros números impares.
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 23 de noviembre de 2023
```

WUOLAH

```
# -----
```

```
#Pedimos los datos
```

```
print("Introduzca el numero de numeros impares que quiere sumar:")
```

```
n<-scan(n=1)
```

```
suma<-0
```

```
numero<-1
```

```
contador<-1
```

```
while(contador<=n){
```

```
  suma<-suma+numero
```

```
  numero<-numero+2
```

```
  contador<-contador+1
```

```
}
```

```
cat("La suma de los primeros", n, "numeros impares sera", suma)
```

Ejercicio B.

Utilizando while

```
# ----
```

```
#' Archivo: p5_parte3_ejb.R
```

```
#' @description Implementar un algoritmo que calcule la suma de los N primeros números naturales. Hacer con for, y while
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 23 de noviembre de 2023
```

```
# -----
```

```
#Pedimos los datos
```

```
print("¿Cuántos números naturales quieres sumar?")
```

```
n<-scan(n=1)
```

```
sum<-0
```

```
contador<-1
```

```
regla<- ((1+n)*n)/2
```

```
while(contador<=n){
```

```
  sum<-sum+ contador
```

```
  contador<-contador+1
```

```
}
```

```
cat("El sumatorio de los", n, "primeros números será", sum,"\n")
```

```
if(regla==sum){
```

```
  cat("Cumple la regla:((1+n)*n)/2=", regla)
```

```
}
```

```
if(regla!=sum){
```

```
  cat("No cumple la regla:((1+n)*n)/2=", regla)
```

```
}
```

Con la función for

```
# ---- -----
```

```
#' Archivo: p5_parte3_ejb.R
```

```
#' @description Implementar un algoritmo que calcule la suma de los N primeros números naturales. Hacer con for, y while
```

```
#' @author Paula Ferrer Ordoñez
```



```
#' Fecha: 23 de noviembre de 2023
```

```
# -----
```

```
#Pedimos los datos
```

```
print("¿Cuántos números naturales quieres sumar?")
```

```
n<-scan(n=1)
```

```
sum<-0
```

```
contador<-1
```

```
regla<- ((1+n)*n)/2
```

```
for (i in 1:n) {
```

```
  sum<-sum+ i
```

```
}
```

```
cat("El sumatorio de los", n, "primeros números será", sum,"\n")
```

```
if(regla==sum){
```

```
  cat("Cumple la regla:((1+n)*n)/2=", regla)
```

```
}
```

```
if(regla!=sum){
```

```
  cat("No cumple la regla:((1+n)*n)/2=", regla)
```

```
}
```

```
Ejercicio C.
```

```
# ---- -----
```

```
#' Archivo: p5_parte3_ejc.R
```

```
#' @description Implementar un algoritmo para calcular el factorial de un número sin usar for
```

```
#' @author Paula Ferrer Ordoñez
```



```
#' Fecha: 23 de noviembre de 2023
```

```
# -----
```

```
#Pedimos los datos
```

```
print("Introduzca el numero del cual quiere realizar el factorial:")
```

```
n<-scan(n=1)
```

```
#Para calcular el factorial tenemos que multipliar todos los numeros hasta n
```

```
numero<- n
```

```
factorial<-1
```

```
#El factorial del numero es la multiplicacion de todos los numeros hasta llegar a este, nosotros lo haremos al reves
```

```
while(numero>=1){
```

```
  factorial<-factorial*numero
```

```
  numero<-numero-1
```

```
}
```

```
cat("El factorial de", n, "sera", factorial)
```

```
Ejercicio D.
```

```
# ---- -----
```

```
#' Archivo: p5_parte3_ejd.R
```

```
#' @description Calcule los N primeros números de la sucesión de Fibonacci y muéstrellos por pantalla.
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 23 de noviembre de 2023
```

```
# -----
```

```

#Pedimos datos

print("incluyendo 0 como numero de la sucesio \n")

print("¿Cuántos números de la sección de fibonacci quiere saber?")

n<- scan(n=1)


vf<-c(0)

a<-0

b<-1

if(n<3){

if(n==1){

  vf<-c(0)

  print(vf)

}

if(n==2){

  vf<-c(0,1)

  print(vf)

}

} else{

for (i in 1:(n-1)) {

  c<- a +b

  a<- b

  b<-c

  vf2<- c(c)

  vf<- c(vf,vf2)

}

cat("Los primeros", n, "números de la sucesión de fibonacci serán", vf)

}

```

Ejercicio E.

```
# -----
```

```
#' Archivo: p5_parte3_eje.R
```

```
#' @description Muestre toda la tabla de multiplicar desde el 1 hasta el N
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 23 de noviembre de 2023
```

```
# -----
```

```
#Pedimos N
```

```
print("Hasta que numero quiere ver las tablas de multiplicar:")
```

```
n<- scan(n=1)
```

```
#Realizamos la multiplicacion de cada numero y mostramos la tabla de ese numero
```

```
if(n<=0){
```

```
  stop("Solo mostrara las tablas desde el uno")
```

```
}
```

```
for(i in 1:n){
```

```
  cero<- 0*i
```

```
  uno<- 1*i
```

```
  dos<- 2*i
```

```
  tres<- 3*i
```

```
  cuatro<- 4*i
```

```
  cinco<- 5*i
```

```
  seis<-6*i
```

```
  siete<- 7*i
```

```
  ocho<- 8*i
```

```
  nueve<- 9*i
```

```
  diez<- 10*i
```

```

cat("La tabla del",i, "\n",i,"x 0=", cero,"\n", i, "x 1=", uno, "\n", i,"x 2=", dos,"\n", i, "x 3=", tres,
"\n", i,"x 4=", cuatro,"\n", i, "x 5=", cinco, "\n",i,"x 6=", seis,"\n", i, "x 7=", siete, "\n", i,"x 8=",
ocho, "\n", i, "x 9=", nueve, "\n", i, "x 10=", diez, "\n")
}

```

Ejercicio F.

```
# ----
```

```
#' Archivo: p5_parte3_eje.R
```

```
#' @description Escriba un programa que permita determinar si un número es primo
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 23 de noviembre de 2023
```

```
# -----
```

```
#Pedimos los datos
```

```
print("Escriba el numero que quiera saber si es primo o no:")
```

```
n<- scan(n=1)
```

```
if(n<=0){
```

```
  stop("El numero debe ser mayor que 0")
```

```
}
```

```
#El 1 y el 2 son primos
```

```
if(n<=1 | n<=2){
```

```
  print("El numero es primo")
```

```
} else{
```

```
# un número es primo si y sólo si no tiene divisores (distintos de la unidad) menores que su raíz cuadrada.
```

```
primo<- 0
```

```
for(d in 2:(sqrt(n))){
```

```

resto<- n%%i

if(resto==0){
  primo<-1
}

}

if(primo==0){
  print("El numero es primo")
} else{
  print("El numero no es primo")
}}

```

Ejercicio G.

```

# ---- -----
#' Archivo: p5_parte3_ejg.R
#' @description Escriba un programa que permita jugar a adivinar un número
#' @author Paula Ferrer Ordoñez
#' Fecha: 23 de noviembre de 2023
# -----

print("Voy a pensar un numero del 1 al 100...")

MAXNUM <- 100; #valor maximo
secreto <- sample(1:MAXNUM,1);

print("Adivina el numero que he pensado:")
n<-scan(n=1)

```

```

while(n!=secreto){
  if(n>secreto){
    cat(n,"es mayor que el numero secreto \n")
    print("Vuelve a introducir otro numero")
    n<- scan(n=1)
  }
  if(n<secreto){
    cat(n,"es menor que el numero secreto \n")
    print("Vuelve a introducir otro numero")
    n<- scan(n=1)
  }
}
if(n==secreto){
  cat("Enhorabuena!! El numero secreto era", secreto)
}

```

EJERCICIO FINAL.

The screenshot shows a web browser displaying a Quarto document titled "Tablas multiplicación" by Paula Ferrer Ordóñez GMA, dated November 2023. The document includes an exercise "Ejercicio E." with the instruction "Muestre toda la tabla de multiplicar desde el 1 hasta N". The R code is presented in several chunks, each with a "Hide" button. The code includes comments in Spanish and uses the `knitr::read_chunk` function to load external code files. It prompts the user for a number 'N' and then generates multiplication tables for numbers from 1 to N.

```

knitr::read_chunk("p5_parte3_eje.R")

#Pedimos N
print("Hasta que numero quiere ver las tablas de multiplicar:")

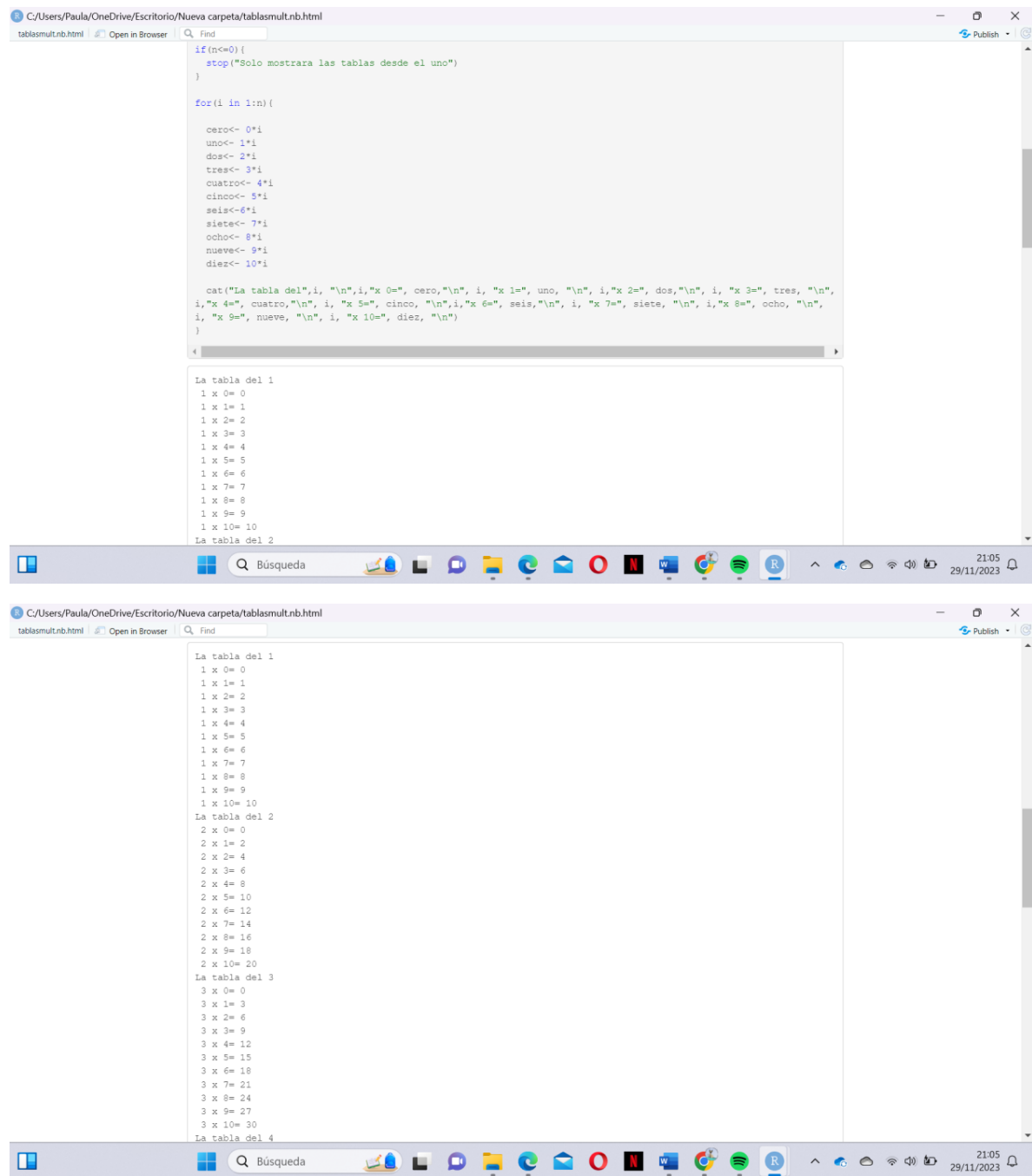
[1] "Hasta que numero quiere ver las tablas de multiplicar:"

n<- scan(n=1)
10

Read 1 item

#Realizamos la multiplicacion de cada numero y mostramos la tabla de ese numero
if (n<=0) {
  stop("Solo mostrara las tablas desde el uno")
}

```



Antes de poder abrirlo deberemos introducir el número para poder ejecutar el algoritmo

Ejercicios Parte 4.

En esta cuarta parte de la practica el objetivo es aprender a manejar los vectores y sus operaciones dentro de Rstudio.


```
-----  
#' Archivo: p5_parte4_ejb.R  
#' @description Obtener la desviación típica de un vector de números V  
#' @author Paula Ferrer Ordoñez  
#' Fecha: 9 noviembre 2023  
#-----
```

```
#Pedimos el vector  
print("Introduzca el vector del que quiere calcular la desviacion tipica")  
v<-scan()  
#Hacemos el vector  
v1<-c(v)  
  
desv<- sd(v1)  
  
#Mostramos los resultados  
  
cat("La desviacion tipica del vector", v1, "es", desv)
```

Ejercicio A.

```
# -----  
#' Archivo: p5_parte4_eja.R  
#' @description Obtener la media de un vector de números V.  
#' @author Paula Ferrer Ordoñez  
#' Fecha: 9 noviembre 2023  
#-----
```

```
#Pedimos el vector  
print("Introduzca el vector del que quiere calcular la media")  
v<-scan()  
#Hacemos el vector
```

```

v1<-c(v)

#Sumamos todos los numeros del vector y lo dividimos entre la longitud del vector
media<- sum(v1)/length(v1)

#Mostramos los resultados

cat("La media de los números del vector", v1, "es",media)

```

Ejercicio B.

```
#
```

Ejercicio C.

```

# -----
#' Archivo: p5_parte4_ejc.R
#' @description Obtener el valor máximo de un vector V y su posición dentro de dicho vector.
#' @author Paula Ferrer Ordoñez
#' Fecha: 9 noviembre 2023
#-----

```

```

#Pedimos el vector
print("Introduzca el vector del que quiere saber el valor maximo")
v<-scan()
#Hacemos el vector
v1<-c(v)
vmax<- 0
for(i in 1:length(v1)){
  if(v1[i]>vmax) {
    vmax<- v1[i]
    pos<- i
  }
}

```

#Mostramos los resultados

```
cat("El valor maximo del vector", v1, "es", vmax, "y su posicion es", pos)
```

Ejercicio D

```
# -----
```

```
#' Archivo: p5_parte4_ejd.R
```

```
#' @description Obtener el valor mínimo de un vector V y su posición dentro de dicho vector.
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 9 noviembre 2023
```

```
#-----
```

```
#Pedimos el vector
```

```
print("Introduzca el vector del que quiere saber el valor minimo y su posicion")
```

```
v<-scan()
```

```
#Hacemos el vector
```

```
v1<-c(v)
```

```
vmin<- min(v1)
```

```
for(i in 1:length(v1)){
```

```
  if(v1[i]==vmin) {
```

```
    vmin<- v1[i]
```

```
    pos<- i
```

```
  }
```

```
}
```

```
#Mostramos los resultados
```

```
cat("El valor minimo del vector", v1, "es", vmin, "y su posicion es", pos)
```

Ejercicio E.

```

# -----

#' Archivo: p5_parte4_eje.R

#' @description Se desea crear un vector de un tamaño seleccionado por el usuario, donde en
cada posición se almacene el valor del factorial del número correspondiente a esa posición

#' @author Paula Ferrer Ordoñez

#' Fecha: 9 noviembre 2023

#-----


#Pedimos los datos


print("Cuantos numeros quiere que tenga el vector?")


n<- scan(n=1)


v<- matrix(1, 1, n)


if(n<1){
  stop("El vector debe tener al menos un numero")
}


#El factorial de 1 es 1


if(n==1) {

  v<- 1
}


if(n>1){

```

#Multiplicamos la posición del vector en el que nos encontramos por el numero anterior, que sera el factorial de la posición anterior

```
for(i in 2:n){  
  
  v[i]<- v[i-1]*i  
}  
}  
print(v)
```

Ejercicio F.

```
# -----  
#' Archivo: p5_parte4_ejf.R  
#' @description Escriba un programa que almacene en un vector todos los números impares  
comprendidos entre dos enteros introducidos por el usuario.  
#' @author Paula Ferrer Ordoñez  
#' Fecha: 9 noviembre 2023  
#-----  
  
#Pedimos los datos  
  
print("Vamos a calcular un vector con todos los numeros impares entre dos numeros")  
  
print("Introduzca el valor menor")  
x<- scan(n=1)  
  
print("Introduzca el valor mayor")  
y<- scan(n=1)  
  
if(x>y){  
  stop("Ha introducido los numeros al reves")
```

```
}
```

```
v<-c()
```

```
#Si el resto del numero entre dos es distinto a 0 será un numero impar
```

```
for(i in x:y){
```

```
  if(i%%2!=0){
```

```
    v<-c(v,i)
```

```
  }
```

```
}
```

```
print(v)
```

Ejercicio G.

```
# -----
```

```
#' Archivo: p5_parte4_ejg.R
```

```
#' @description Calcular la diferencia entre los elementos mayor y menor de un vector V.
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 9 noviembre 2023
```

```
#-----
```

```
#Pedimos el vector
```

```
print("Introduzca el vector")
```

```
n<- scan()
```

```
v<-c(n)
```

```
#Buscamos el valor menor y mayor
```

```
max<- max(v)
```

```
min<- min(v)
```

```
#Hacemos la diferencia
```

```
dif<- max-min
```

```
cat("La diferencia entre el valor mayor ",max, ", y menor ", min, ", del vector es",dif)
```

Ejercicio H.

```
# -----
```

```
#' Archivo: p5_parte4_ejh.R
```

```
#' @description Escriba un programa que pida un número entero por teclado y guarde en un  
vector su tabla de multiplicar del 1 al 9. Muestre por pantalla el contenido del vector generado  
como si fuese la tabla de multiplicar.
```

```
#' @author Paula Ferrer Ordoñez
```

```
#' Fecha: 9 noviembre 2023
```

```
#-----
```

```
#Pedimos los datos
```

```
print("De que numero quiere la tabla de multiplicar:")
```

```
n<- scan(n=1)
```

```
tabla<- 1:9
```

```
cat("----- \n")
```

```
cat("La tabla del", n, "\n")
```

```
cat("----- \n")
```

```
for(i in 1:9){
```

```
  tabla[i]<- n*i
```

```

cat(n," x", i, "=", tabla[i], "\n")
}

```

Ejercicios parte 5.

En esta parte aprenderemos a crear funciones en Rstudio que podamos utilizar en nuestros scripts.

Para ellos debemos crear un script con el mismo nombre que tenga la función y seguir el siguiente modelo:

```

Nombre_de_la_función<- function(variables de entrada) {
  y<- m * x + n;
  return(y);
}

```

Donde y sería el resultado de la función.

En las funciones no puede haber datos de salida, devolverá el valor que se quede guardada en return cuando en otro script lo pongamos como un dato de salida, por ejemplo cat(es_par(n))

Ejercicio A.

```

# -----
#' @title es_par
#' @description Función es_par(n) que devuelve el valor 1 sólo si el número n pasado como
parámetro es par y 0 en otro caso.
#' @param n numero que queremos saber si es par
#' @return 1 si es par, 0 si no lo es
# -----

es_par<- function(n){

  par<-0

  if(n%%2==0){
    par<-1
  }
}

```



```

    } else{
        cat(par)
    }
return(par);
}

```

Ejercicio B.

```

# -----

#' @title es_neg

#' @description Función es_negativo(n) que devuelve el valor 1 sólo si el número n pasado
como parámetro es negativo y 0 en otro caso.

#' @param n numero que queremos saber si es positivo

#' @return 1 si es positivo, 0 si es 0 o negativo

# -----

es_neg<- function(n){
  neg<- 0
  if(n>0){
    neg<- 1
  }
  return(neg)
}

```

Ejercicio C.

```

# -----

#' @title enmedio

#' @description Función enmedio(a,b,c) que devuelve el valor 1 sólo si el número b está entre
a y c, 0 en otro caso.

#' @param a numero más pequeño

#' @param b numero que queremos saber si está entre medias

#' @param c numero mayor

#' @return 1 si esta entre medias, 0 sino lo está

# -----

```

```

enmedio<- function(a,b,c){
  med<- 0
  if(a<b){
    if(b<c){
      med<-1
    }
  }
}

```

```

return(med)
}

```

Ejercicio D.

```

# -----

```

```

#' @title mayor

```

```

#' @description Función mayor(a,b,c) que devuelve el valor mayor entre a,b,c. Debe hacer uso
de la estructura if.

```

```

#' @param a numero

```

```

#' @param b numero

```

```

#' @param c numero

```

```

#' @return el numero mayor

```

```

# -----

```

```

mayor<-function(a,b,c)

```

```

{
  if(a>b){
    if(a>c){
      mayor<-a
    }else {
      mayor<-c
    }
  } else{

```

```

    if (b>c){
      mayor<-b
    } else{
      mayor<-c
    }
  }
  return(mayor)
}

```

Ejercicio E.

```

# -----
#' @title ecuacion_parabola
#' @description Función ecuacion_parabola(x,a,b,c): aplica a un valor x la ecuación de la
parábola  $y = ax^2+bx+c$ 
#' @param x valor para el cual queremos calcular la ecuación
#' @param a valor que se multiplica a  $x^2$ 
#' @param b valor que se multiplica a x
#' @param c termino independiente
#' @return el numero mayor
# -----

```

```

ecuacion_parabola<- function(x,a,b,c){
  y<- a*x*x+ b*x+ c

  return(y)
}

```

Ejercicio F.

```

# -----
#' @title miexponencial
#' @description f(x)
#' @param x variable
# -----

```

```
miexponencial<- function(x){
  y<- (1/(sqrt(2*pi)))*exp((-1/2)*x*x)
}
```

```
return(y)
```

Ejercicio G.

```
# -----
#' @title miTrigonometria
#' @description r(o)
#' @param o radianes para los que queremos calcular la función
# -----
```

```
miTrigonometria<- function(o){
  r<- exp(sin(o))-2*cos(40)+sin(((20- pi)/24))^5

  return(r)
}
```

Ejercicio H.

TEST ECUACIÓN DE LA PARABOLA

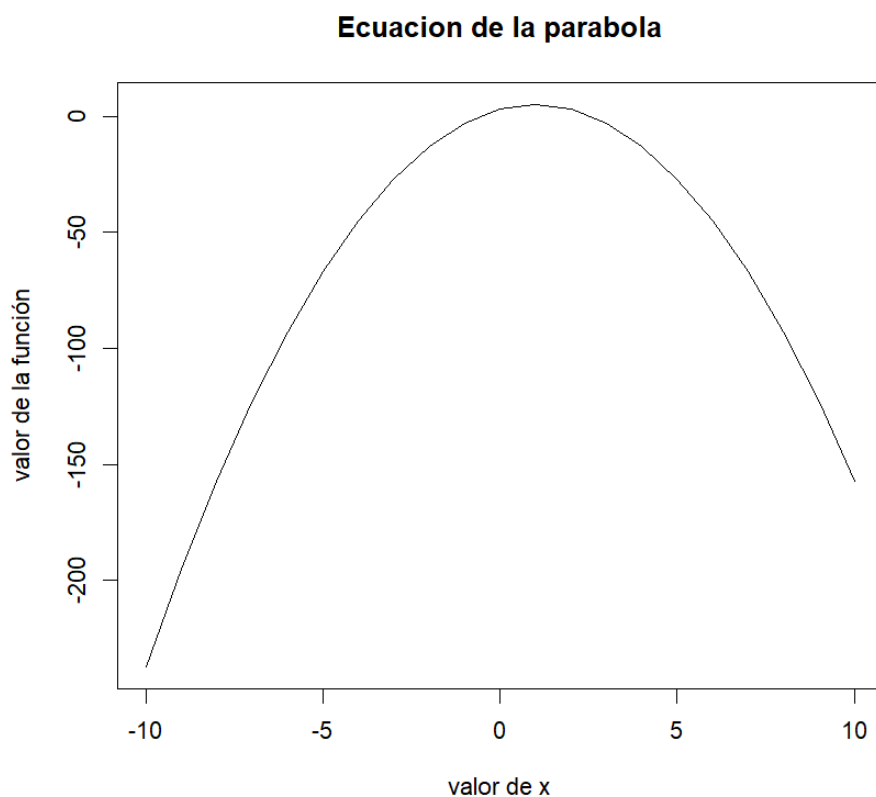
```
source("ecuacion_parabola.R")
print("Introduzca el valor que se multiplica a x^2")
a<- scan(n=1)
print("Introduzca el valor que se multiplica a x")
b<- scan(n=1)
print("Introduzca el termino independiente")
c<- scan(n=1)

y<-ecuacion_parabola(x,a,b,c)
```

```
x<-c(-10:10)
```

```
plot(x, y, "l", main = "Ecuacion de la parabola", xlab = "valor de x", ylab = "valor de la función")
```

```
[1] "Introduzca el valor que se multiplica a x^2"  
1: -2  
Read 1 item  
[1] "Introduzca el valor que se multiplica a x"  
1: 4  
Read 1 item  
[1] "Introduzca el termino indepedeinte"  
1: 3  
Read 1 item
```



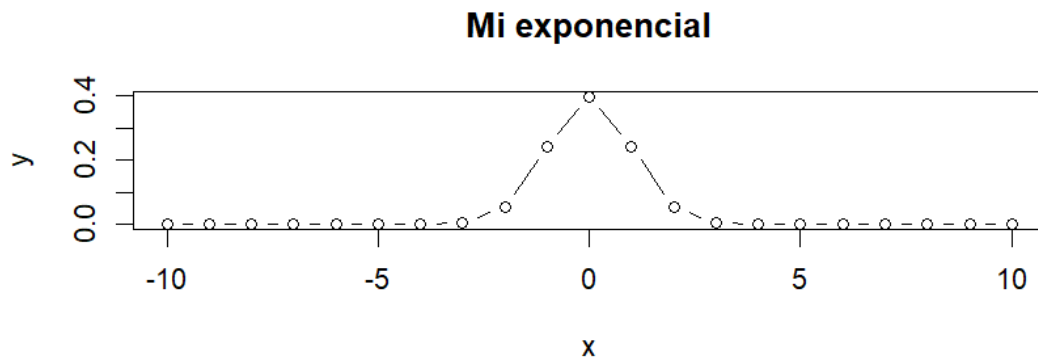
TEST FUNCIÓN MI EXPONENCIAL

```
source("miexponencial.R")
```

```
x<-c(-10:10)
```

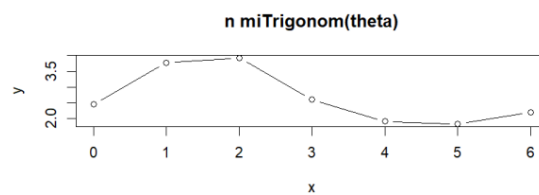
```
y<- miexponencial(x)
```

```
plot(x,y,type="b",main = "Mi exponencial")
```



TEST ECUACION TRIGONOMICA

```
source("miTrigonom.R")
x<- c(0:(2*pi))
y<- miTrigonom(x)
plot(x,y, type="b", main="n miTrigonom(theta)")
```



Ejercicio i.

title: "Ejercicio E"

output: html_notebook

****Función ecuacion_parabola(x,a,b,c): aplica a un valor x la ecuación de la parábola:****

$$y = \frac{1}{\sqrt{2\pi}} e^{\frac{-1}{2}x^2}$$

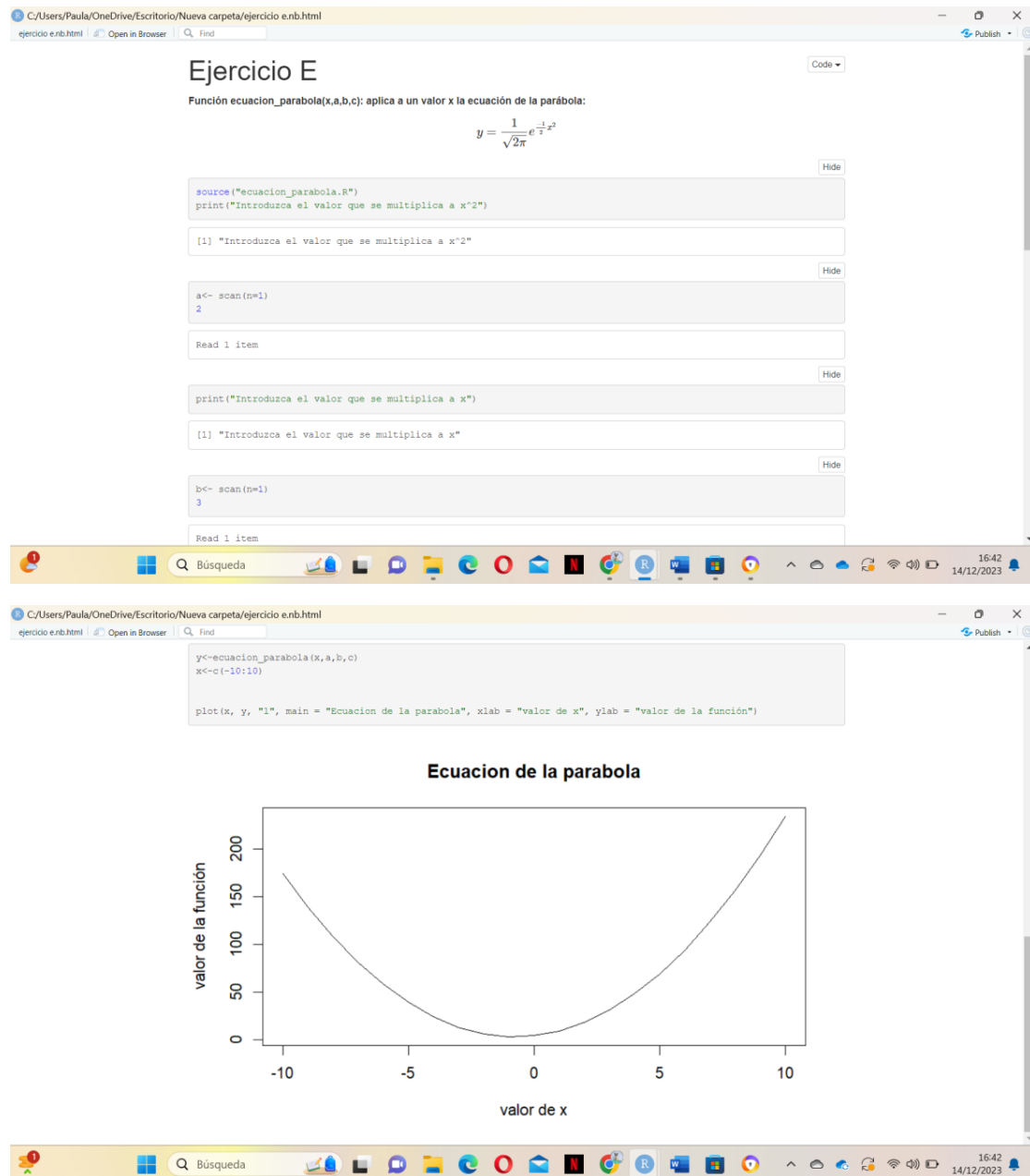
```
```{r setup, echo= FALSE}
```

```
knitr::read_chunk("test_ecuacion_parabola.R")
```

```
```
```

``{r chunk}

``



Ejercicios parte 6

El objetivo de esta práctica es la de aprender a utilizar y trabajar con cadenas de caracteres en Rstudio y usar ficheros de datos en R.

Las funciones que utilizaremos en esta parte son:

nchar(str). Nos dirá la longitud de la cadena de texto

Unique(v). Nos mostrará los valores que no se repiten de un vector

strsplit(cad1,"")[[1]]. Separa los valores de la cadena en elementos de un vector, el primer valor hace referencia a lo que queremos separar y el segundo de qué forma, en este caso "" significa que se separará en cada elemento.

paste0(cad2v,collapse="")

Ejercicio A.

```
cad1<- "reconocer"
```

```
cad2<- "dabale arroz a la zorra el abad"
```

```
cad3<- "se van sus naves"
```

```
cad4<- "ATCGGCTAATCGGCTA"
```

#La función nchar(strng) nos dice la longitud de la cadena

```
cat("La longitud de la cadena 1 es", nchar(cad1), ", la longitud de la cadena 2 es", nchar(cad2),  
", la longitud de la cadena 3 es", nchar(cad3), "y la longitud de la cadena 4 es", nchar(cad4))
```

La longitud de la cadena 1 es 9 , la longitud de la cadena 2 es 31 , la longitud de la cadena 3 es 16 y la longitud de la cadena 4 es 16

Ejercicio B.

Mostrar cuántos caracteres diferentes aparecen en cada cadena.

#Primero debemos colocar las cadenas

```
cad1<- "reconocer"
```

```
cad2<- "dabale arroz a la zorra el abad"
```

```
cad3<- "se van sus naves"
```

```
cad4<- "ATCGGCTAATCGGCTA"
```

#Separamos la cadena para tener cada elemento en un vector

```
v1<-strsplit(cad1,"")[[1]]
```

#Eliminamos los valores repetidos

```
valun1<- unique(v1)
```

```
cat("Los diferentes valores que hay en la cadena 1 son \n", valun1, "\n")
```

#Lo repetimos con el resto de cadenas

```
v2<-strsplit(cad2,"")[[1]]
```

```
valun2<- unique(v2)
```

```
cat("Los diferentes valores que hay en la cadena 2 son:\n", valun2, "\n")
```

```
v3<-strsplit(cad3,"")[[1]]
```

```
valun3<- unique(v3)
```

```
cat("Los diferentes valores que hay en la cadena 3 son: \n", valun3, "\n")
```

```
v4<-strsplit(cad4,"")[[1]]
```

```
valun4<- unique(v4)
```

```
cat("Los diferentes valores que hay en la cadena 4 son:\n", valun4, "\n")
```


Los diferentes valores que hay en la cadena 1 son
 r e c o n
 Los diferentes valores que hay en la cadena 2 son:
 d a b l e r o z
 Los diferentes valores que hay en la cadena 3 son:
 s e v a n u
 Los diferentes valores que hay en la cadena 4 son:
 A T C G

Ejercicio C.

#Almacene en un vector, y muestre por pantalla, el histograma (frecuencia absoluta) de aparición de cada carácter en la cadena cad2. No es necesaria una representación gráfica.

#Escribimos la cadena

```
cad2<- "dabale arroz a la zorra el abad"
```

#Separamos la cadena por elementos y almacenamos en un vector
 v<-strsplit(cad2,"")[[1]]

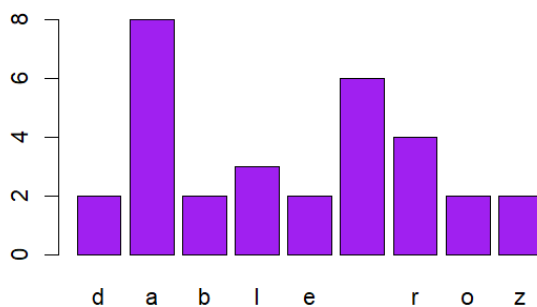
#Creamos otro vector con elementos sin repetir

```
v2<-unique(v)
cont<-c()
```

```
for (i in 1:length(v2)) {
  foundL <- gregexpr(v2[i], cad2) #Buscamos cuantas veces se repite en cad2 v2[i]
  found <- foundL[[1]]
  long <- length(found)
  cont <- c(cont, long)
}
```

```
data <- rbind(v2, cont) #Une los dos vectores, veces y u2 asociando los valores de uno a otro.
print(data)
```

```
barplot(cont, names.arg = v2, col="purple")
```



Ejercicio D.

#Elimine de forma automática los espacios en blanco de las cadenas cad2 y cad3.

```
cad2<- "dabale arroz a la zorra el abad"
```

```
cad3<- "se van sus naves"
```

#convertimos las cadenas en vectores

```
cad2v<- strsplit(cad2," ")[[1]] #[[1]] saca de la lista los elementos
```

```
cad3v<- strsplit(cad3," ")[[1]] #[[1]] saca de la lista los elementos
```

```
cad2new<- paste0(cad2v,collapse="") #collapes="" es necesario para que la cadena se quede  
unida sin espacios
```

```
cad3new<- paste0(cad3v,collapse="")
```

```
cat(cad2,"\n sin espacios seria: \n", cad2new,"\n", cad3, "\n sin espacios seria:\n", cad3new)
```

Ejercicio E.

#En la cadena cad4, intercambie de forma automática los símbolos A por T y T por A. Es decir, dada la cadena "ACTG" obtendríamos "TCAG".

```
cad4<- "ATCGGCTAATCGGCTA"
```

```
cad4n<-gsub("A","X",cad4) #Cambiamos las As de la cadena 4 por Xs para que al cambiar las Ts  
por As no sea un cambio nulo
```

```
cad4n<-gsub("T","A", cad4n) #Cambiamos las Ts por As
```

```
cad4n<-gsub("X","T", cad4n) #Cambiamos las xs por ts
```

```
cat(cad4,"cambiando las As por las Ts y viceversa seria:\n",cad4n )
```

Ejercicio F.

#Realice una función llamada esPalindromo(cad) que devuelva true si la cadena cad es un palíndromo (i.e. se lee igual de izquierda a derecha que de derecha a izquierda) y false en otro caso.

```
palindromo<-function(cad){  
  pal<-FALSE  
  cadv<- strsplit(cad," ")[[1]]  
  cadA<- paste0(cadv, collapse = "")  
  cadv2<- rev(cadv)  
  cadn<- paste0(cadv2, collapse = "")  
  
  if(cadA==cadn){  
    pal<- TRUE  
  }  
  return(pal)  
}
```

uso esta función, compruebe de forma automática si las cadenas cad1, cad2, cad3, cad4 son palíndromos. IMPORTANTE: debe eliminar los espacios en blanco de las cadenas.

```
source("palindromo.R")
```

```
cad1<- "reconocer"  
cad2<- "dabale arroz a la zorra el abad"  
cad3<- "se van sus naves"  
cad4<- "ATCGGCTAATCGGCTA"
```

```
cat( cad1,"es", print(palindromo(cad1)),"\n", cad2,"es", print(palindromo(cad2)),"\n",  
cad3,"es", print(palindromo(cad3)), "\n", cad4,"es", print(palindromo(cad4)))
```

Ejercicios parte 7.

En esta parte de la práctica aprenderemos a utilizar los diferentes comandos que nos ofrece Rstudio para llevar a cabo la representación de graficas.

Las funciones utilizadas en los ejercicios son:

Plot(x,y). Utilizaremos este comando para generar una grafica con dos vectores numéricos. Dentro de esta función tenemos opciones para ponerle título(**main=**), cambiar el tipo de línea (**type=**), el color (**col=**), nombre de los ejes y (**ylab=**) y de los ejes x (**xlab=**), también podremos determinar los valores entre los que representaremos y con (**ylim=**), y añadir otra línea con otros valores de y con **points(x,y)**, teniendo las mismas opciones que con plot.

Png("nombre.png"). Crear un archivo png donde podamos guardar nuestra grafica. Hay que ponerlo antes de crear nuestra grafica.

Dev.off(). Guarda la grafica en nuestro archivo png. Hay que colocarlo después de crear nuestra grafica.

Barplot(x). Crea un grafico de barras con nuestro vector numérico, tiene las mismas opciones que plot.

Legend. Crea una leyenda para nuestro grafico, podemos elegir en que zona del grafico colocarlo por ejemplo con **bottomleft** o similares, utilizaremos dentro **legend=c("nombre 1", nombre 2", "etc")**, para que la leyenda tenga los nombres de las funciones, podemos asignarle un color con **col=c("color1", "color2","etc")**, para que salga al lado del nombre una línea de color hay que poner **lty=1**, hay otras opciones que podemos ver en el apartado help

Pie(x). Crea una grafica en forma pie chart con un vector numérico, tienes diferentes opciones de personalización.

Ejercicio 1.

#Gráfica de la función $f(x) = \sin(x) + \cos(x)$ en el intervalo $[-2\pi, 2\pi]$ en pasos de 0.1. Use el color rojo.

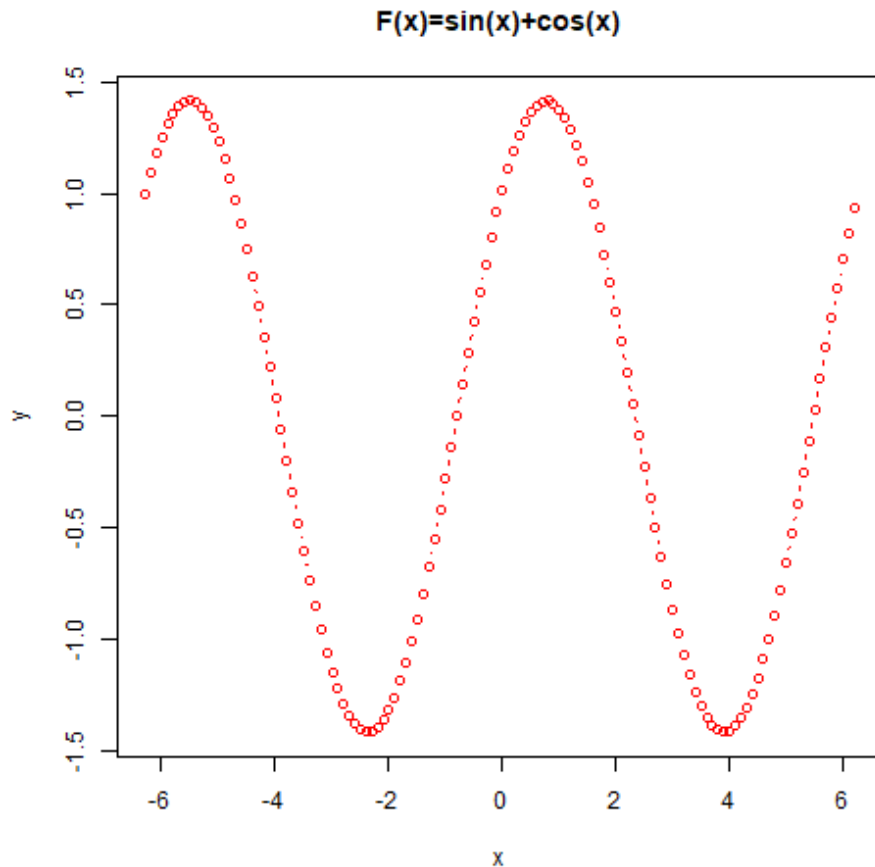
```
x<- c(seq((-2*pi),(2*pi),0.1)) #queremos que vaya desde -2pi a 2pi con pasos de 0.1
```

```
y<- sin(x) + cos(x)
```

```
plot(x,y, main="F(x)=sin(x)+cos(x)", col="red", type="b")
```

```
png("sin(x)+cos(x).png")
```

```
dev.off()
```



Ejercicio B.

#Gráfica de la función en el intervalo [-1,1] en pasos de 0.05. Use el color verde.

png ("ejb.png")

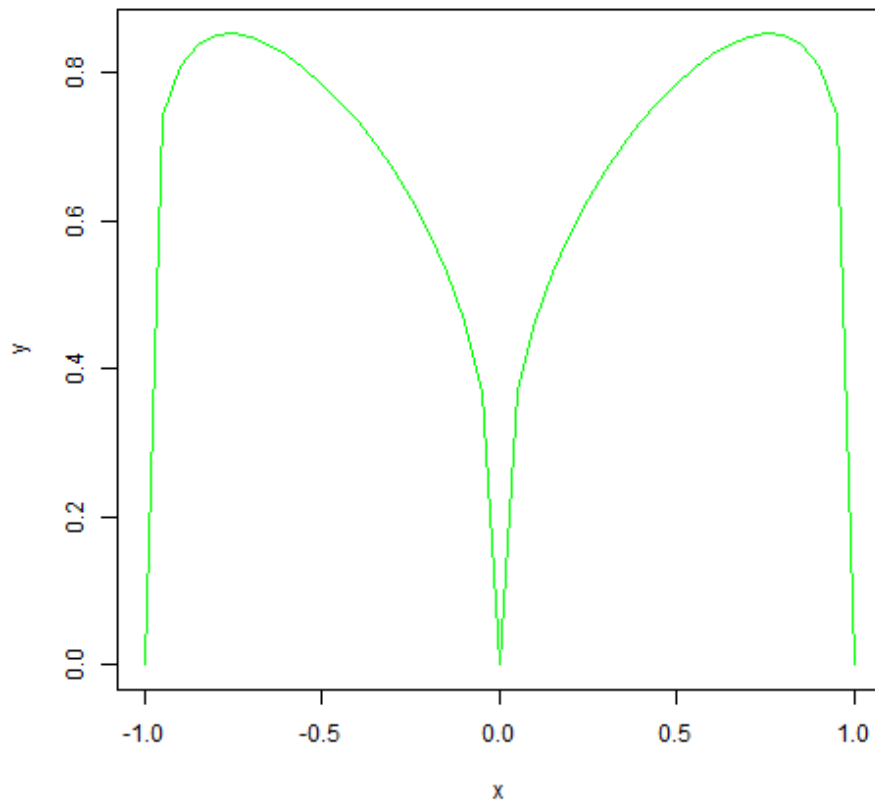
x<-seq(-1,1,0.05) #creamos un vector con números del -1 al 1 con pasos de 0.5

y<-(x^2-x^6)^(1/6)

plot(x,y,"I",main="Ejercicio B", col="green")

dev.off()

Ejercicio B

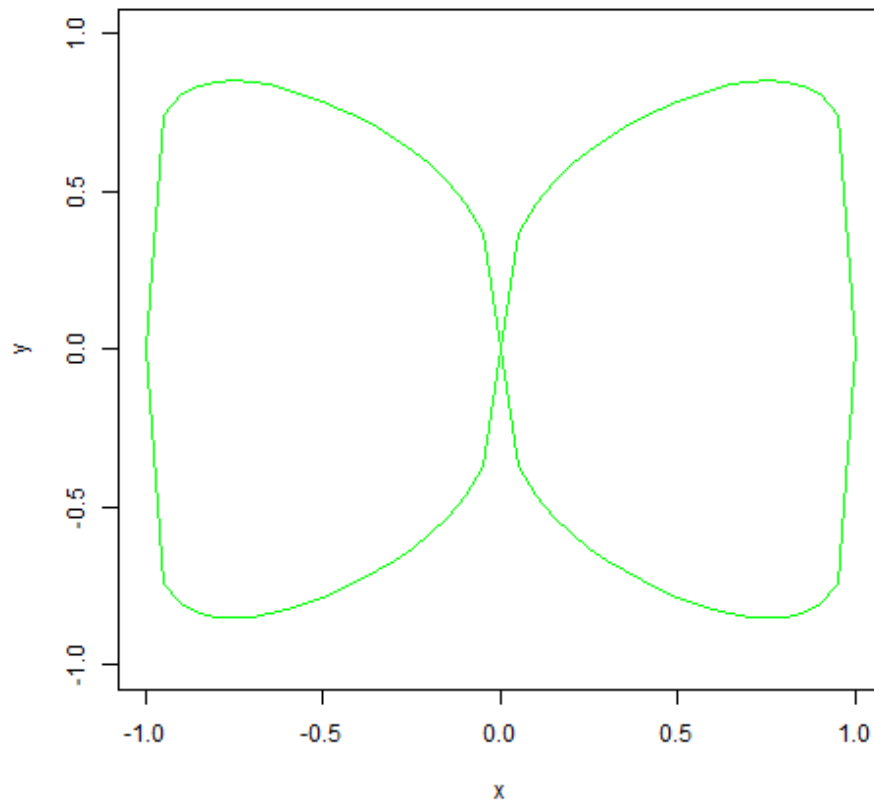


Ejercicio C.

#Añada a la figura anterior (use points), la gráfica de la función $f(x) = -(x^2 - x^6)^{1/6}$ en el intervalo $[-1, 1]$ en pasos de 0.05. Use el color verde.

```
png("ejc.png")
x<-seq(-1,1,0.05)
y<-(x^2-x^6)^(1/6)
y2<- -(x^2-x^6)^(1/6)
plot(x,y,"l",main="Ejercicio c", col="green", ylim=c(-1,1))
points(x,y2,type="l",col="green")
```

Ejercicio c



dev.off()

Ejercicio D.

png("ejd.png")

```
x<- c(seq(-2,2,0.1))
```

sigma1<- 1 #Damos 3 valores diferentes a sigma con sigma1,sigma2,sigma3, repitiendo la función con cada valor

```
y<- (-1/(pi*(sigma1)^4))*(1-((x^2)/(2*(sigma1^2))))*exp(-(x^2)/(2*(sigma1)^2))
```

```
sigma2<- 1.25
```

```
y2<- (-1/(pi*(sigma2)^4))*(1-((x^2)/(2*(sigma2^2))))*exp(-(x^2)/(2*(sigma2)^2))
```

```
sigma3<- 1.5
```

```
y3<- (-1/(pi*(sigma3)^4))*(1-((x^2)/(2*(sigma3^2))))*exp(-(x^2)/(2*(sigma3)^2))
```

```
plot(x,y, type="l", main="Ejercicio D", xlab="Valores que toma x", ylab="Valores que toma la  
funcion", col="pink",ylim=c(-0.4,0.1))
```

#Añadimos los puntos de la función con los otros dos valores de sigma

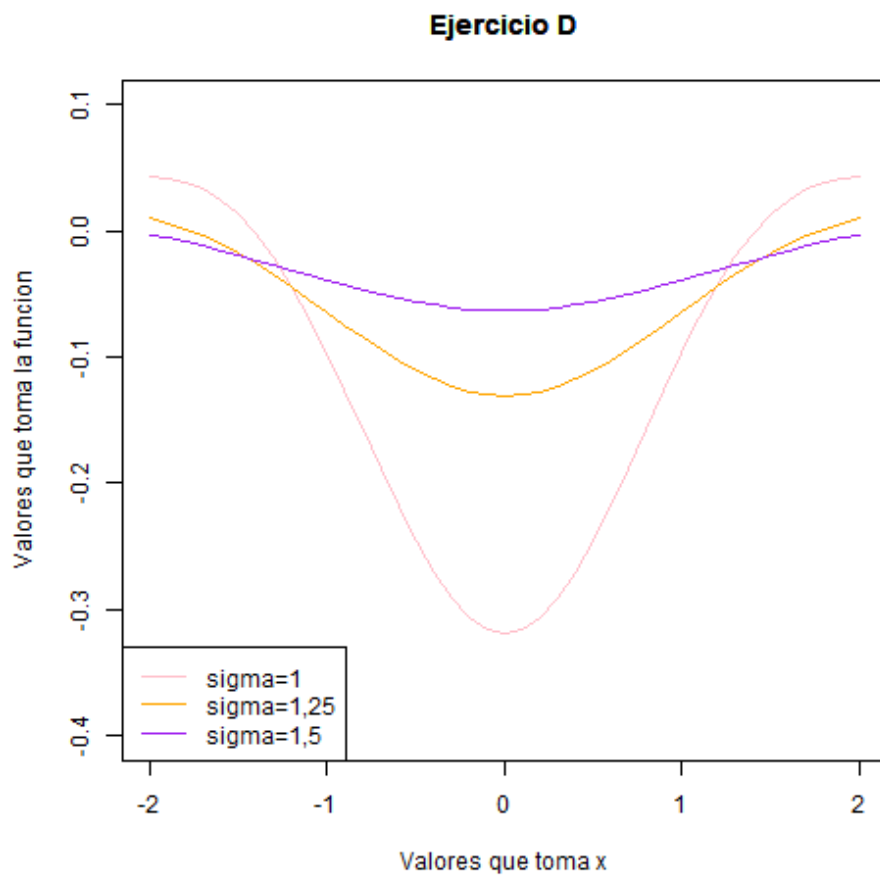
```
points(x,y2, type="l", col="orange")
```

```
points(x,y3,type="l", col="purple")
```

```
title("Ejercicio D")
```

```
legend("bottomleft", legend=c("sigma=1","sigma=1,25","sigma=1,5"),  
col=c("pink","orange","purple"),lty=1)
```

```
dev.off()
```



Ejercicio E.


```
png("eje.png")
```

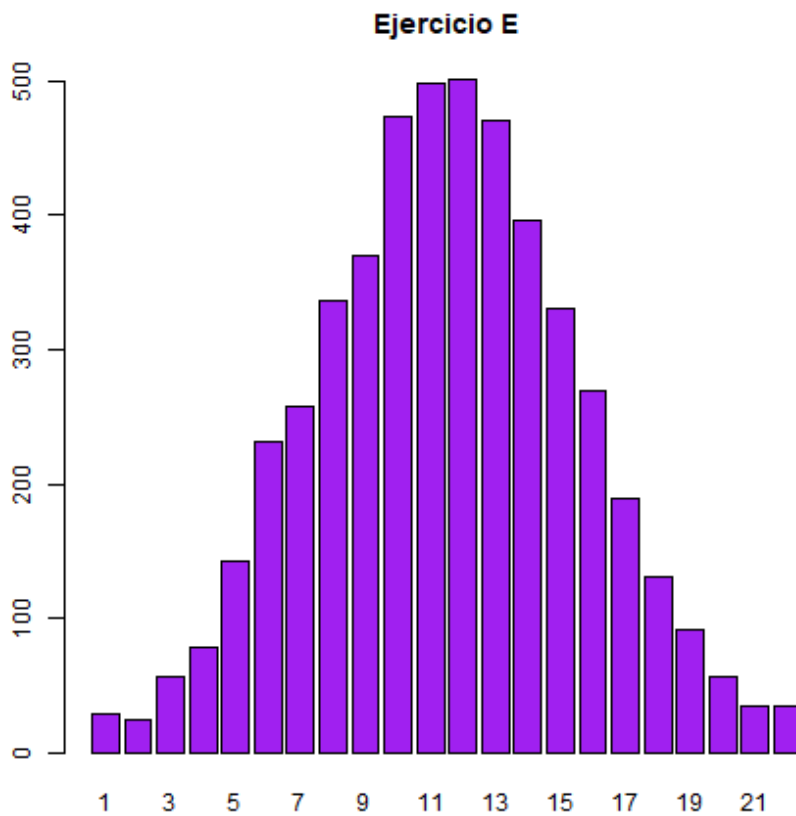
```
h <- hist(rnorm(5000,0,0.4), c(-2,seq(-1,1,0.1),2), plot=FALSE)$counts
```

```
##(rnorm(5000,0,0.4), creará 5000 números aleatorios con distribución normal, media 0 y  
desviación típica de 0.4
```

```
# c(-2,seq(-1,1,0.1),2), creará un vector con, -2, los números del -1 al 1 con pasos de 0,1 y el 2,
```

```
barplot(h, names.arg = (1:length(h)) nm, col="purple", main="Ejercicio E", xlab="Posicion",  
ylab="Frecuencia")
```

```
dev.off()
```



Ejercicio F.

```
png("ejf.png")
```

```
h <- hist(rnorm(5000,0,0.4), c(-2,seq(-1,1,0.25),2))$counts
```

```
pie(h, main="Ejercicio F")  
dev.off()
```

Ejercicio F

