

Examen final: Verificación del tipo de número.

Alan Andrés Mérida Morales, 202100023^{1,*}

¹Facultad de Ingeniería, Escuela de Ingeniería Mecánica Eléctrica,
Universidad de San Carlos, Ciudad Universitaria, Zona 12, Guatemala.

I. INTRODUCCIÓN

Este programa es una aplicación de consola en Python y Octave que permite a los usuarios ingresar números y verificar ciertas propiedades matemáticas de estos números. La aplicación está conectada a una base de datos PostgreSQL para gestionar la información de los usuarios y almacenar los resultados de las verificaciones. Los usuarios pueden registrarse e iniciar sesión con su nombre y carné. Una vez autenticados, pueden ingresar un número y verificar si es primo, perfecto o palíndromo. Los resultados de estas verificaciones se guardan tanto en la base de datos como en un archivo de texto con un formato que incluye el nombre del usuario, su carné, el número ingresado y las propiedades verificadas. Además, la aplicación permite ver el historial de las verificaciones realizadas y ofrece la opción de borrar los registros tanto en la base de datos como en el archivo de texto. El programa proporciona una interfaz de menú interactivo para facilitar la navegación y el uso de las diferentes funcionalidades.

II. RESULTADOS

A. Códigos realizados

1. Python

```
1 import psycopg2
2 import os
3 import re
4
5 # Conexión a la base de datos
6 conn = psycopg2.connect(
7     dbname='user',
8     user='postgres',
9     password='@merida',
10    host='localhost',
11    port='5432'
12 )
13 cursor = conn.cursor()
14
15 # Nombre del archivo de texto
16 archivo_texto = 'resultados_verificaciones.txt'
17
18 # Verificar si el archivo de texto existe, si no, crearlo
19 if not os.path.exists(archivo_texto):
20     with open(archivo_texto, 'w') as archivo:
21         archivo.write('')
22     print('El archivo de texto ha sido creado.')
23 else:
24     print('El archivo de texto ya existe.')
25
26 # Función para validar que no haya caracteres especiales
27 def contiene_caracteres_especiales(cadena):
28     return bool(re.search(r'["a-zA-Z0-9 ]', cadena))
29
30 # Función para pedir una opción numérica
31 def pedir_opcion_numerica(mensaje):
32     while True:
33         opcion = input(mensaje).strip()
34         if opcion.isdigit():
35             return opcion
```

Figura 1: Código realizado en python

```
36 else:
37     print('Por favor ingrese solo números.')
38
39 # Menú inicial para ingresar usuario
40 while True:
41     print('--- Menú Inicial ---')
42     print('1. Ingresar Usuario')
43     print('2. Salir')
44     opcion_inicial = pedir_opcion_numerica('Ingrese su elección: ')
45
46     if opcion_inicial == '2':
47         print('Saliendo del programa...')
48         break
49     elif opcion_inicial == '1':
50         respuesta = input('¿Ya tienes un usuario registrado? (Y/N): ').strip().upper()
51
52         if respuesta == 'Y':
53             nombre = input('Ingrese su nombre: ').strip()
54             carne = input('Ingrese su carné (9 dígitos): ').strip()
55
56             if contiene_caracteres_especiales(nombre):
57                 print('El nombre no debe contener caracteres especiales.')
58                 continue
59
60             if not carne.isdigit() or len(carne) != 9:
61                 print('El carné debe contener exactamente 9 dígitos numéricos.')
62                 continue
63
64             cursor.execute('SELECT id FROM usuario WHERE nombre = %s AND carne = %s',
65                             (nombre, carne))
66             result = cursor.fetchone()
```

Figura 2: Código realizado en python

```
67 if not result:
68     print('Usuario no encontrado. Por favor, registre primero los datos del
69     usuario.')
70     continue
71 else:
72     print('¡Bienvenido de nuevo!')
73
74 while True:
75     print('--- Menú de Usuario ---')
76     print('1. Ingresar datos')
77     print('2. Ver historial de datos')
78     print('3. Borrar datos')
79     print('4. Salir')
80     opcion_usuario = pedir_opcion_numerica('Ingrese su elección: ')
81
82     if opcion_usuario == '1':
83         numero_str = input('Ingrese el número a verificar: ').strip()
84
85         if not numero_str.isdigit():
86             print('Por favor ingrese solo números.')
87         else:
88             numero = int(numero_str)
89
90             # Verificar si es primo
91             es_primo = numero > 1 and all(numero % i != 0 for i in range(2,
92                                     int(numero ** 0.5) + 1))
93
94             if es_primo:
95                 print('El número es primo.')
96             else:
97                 print('El número no es primo.')
98
99             # Verificar si es perfecto
100             suma_divisores = sum(i for i in range(1, numero) if numero % i
101                                 == 0)
102             es_perfecto = suma_divisores == numero
103
104             if es_perfecto:
105                 print('El número es perfecto.')
106             else:
```

Figura 3: Código realizado en python

* 3690273450101@ingenieria.usac.edu.gt

```

182         print('El número no es perfecto.')
183
184     # Verificar si es palindromo
185     es_palindromo = str(numero) == str(numero[::-1])
186     if es_palindromo:
187         print('El número es palindromo.')
188     else:
189         print('El número no es palindromo.')
190
191     # Insertar los resultados en la base de datos
192     cursor.execute('INSERT INTO verificaciones (id_usuario,
193 numero, es_primo, es_perfecto, es_palindromo)
194     VALUES (%s, %s, %s, %s, %s);',
195             (result[0], numero, es_primo, es_perfecto,
196              es_palindromo))
197
198     conn.commit()
199     print('Los resultados han sido guardados en la base de datos.')
200
201     # Guardar en el archivo de texto
202     with open(archivo_texto, 'a') as archivo:
203         archivo.write(f'Usuario: {nombre} (Carné: {carne})\n')
204         archivo.write(f'Número: {numero}\n')
205         archivo.write(f'Es primo: (1 if es_primo else 0)\n')
206         archivo.write(f'Es perfecto: (1 if es_perfecto else 0)\n')
207         archivo.write(f'Es palindromo: (1 if es_palindromo else
208 0)\n')
209
210     print('Datos guardados correctamente.')
211
212     elif opcion_usuario == '2':
213         # Mostrar historial de datos desde la tabla 'verificaciones'
214         cursor.execute('SELECT id_usuario, numero, es_primo, es_perfecto,
215 es_palindromo FROM verificaciones;')
216         result = cursor.fetchall()
217
218         if not result:
219             print('No hay datos registrados.')
220         else:

```

Figura 4: Código realizado en python

```

136         print('----- Historial de verificaciones -----')
137         print('ID Usuario|Número|Primo|Perfecto|Palindromo')
138
139     # Iterar sobre los resultados y mostrarlos
140     for row in result:
141         id_usuario, numero, es_primo, es_perfecto, es_palindromo =
142         row
143         print(f'{id_usuario}|{numero}|{es_primo}|{es_perfecto}|{es_palindromo}')
144
145     elif opcion_usuario == '3':
146         try:
147             # Borrar los registros de la tabla 'verificaciones'
148             cursor.execute('DELETE FROM verificaciones;')
149             conn.commit()
150
151             print('Todos los datos de la tabla de verificaciones han sido
152 borrados.')
153
154             # Borrar los datos del archivo de texto 'resultados_verifica-
155 ciones.txt'
156             if os.path.exists(archivo_texto):
157                 # Si el archivo existe, borrarlo
158                 with open(archivo_texto, 'w') as archivo:
159                     archivo.write('')
160                 print('Los datos del archivo de texto "resultados_verifica-
161 ciones.txt" han sido borrados.')
162             else:
163                 print('El archivo "resultados_verificaciones.txt" no
164 existe.')
165
166         except Exception as e:
167             print(f'Error al borrar los datos: {e}')
168
169     except Exception as e:
170         print(f'Error al borrar los datos: {e}')
171
172     except Exception as e:
173         print(f'Error al borrar los datos: {e}')
174
175     except Exception as e:
176         print(f'Error al borrar los datos: {e}')
177
178     except Exception as e:
179         print(f'Error al borrar los datos: {e}')
180
181     except Exception as e:
182         print(f'Error al borrar los datos: {e}')
183
184     except Exception as e:
185         print(f'Error al borrar los datos: {e}')
186
187     except Exception as e:
188         print(f'Error al borrar los datos: {e}')
189
190     except Exception as e:
191         print(f'Error al borrar los datos: {e}')
192
193     except Exception as e:
194         print(f'Error al borrar los datos: {e}')
195
196     except Exception as e:
197         print(f'Error al borrar los datos: {e}')
198
199     except Exception as e:
200         print(f'Error al borrar los datos: {e}')
201
202     except Exception as e:
203         print(f'Error al borrar los datos: {e}')
204
205     except Exception as e:
206         print(f'Error al borrar los datos: {e}')
207
208     except Exception as e:
209         print(f'Error al borrar los datos: {e}')
210
211     except Exception as e:
212         print(f'Error al borrar los datos: {e}')
213
214     except Exception as e:
215         print(f'Error al borrar los datos: {e}')
216
217     except Exception as e:
218         print(f'Error al borrar los datos: {e}')
219
220     except Exception as e:
221         print(f'Error al borrar los datos: {e}')
222
223     except Exception as e:
224         print(f'Error al borrar los datos: {e}')
225
226     except Exception as e:
227         print(f'Error al borrar los datos: {e}')
228
229     except Exception as e:
230         print(f'Error al borrar los datos: {e}')
231
232     except Exception as e:
233         print(f'Error al borrar los datos: {e}')
234
235     except Exception as e:
236         print(f'Error al borrar los datos: {e}')
237
238     except Exception as e:
239         print(f'Error al borrar los datos: {e}')
240
241     except Exception as e:
242         print(f'Error al borrar los datos: {e}')
243
244     except Exception as e:
245         print(f'Error al borrar los datos: {e}')
246
247     except Exception as e:
248         print(f'Error al borrar los datos: {e}')
249
250     except Exception as e:
251         print(f'Error al borrar los datos: {e}')
252
253     except Exception as e:
254         print(f'Error al borrar los datos: {e}')
255
256     except Exception as e:
257         print(f'Error al borrar los datos: {e}')
258
259     except Exception as e:
260         print(f'Error al borrar los datos: {e}')
261
262     except Exception as e:
263         print(f'Error al borrar los datos: {e}')
264
265     except Exception as e:
266         print(f'Error al borrar los datos: {e}')
267
268     except Exception as e:
269         print(f'Error al borrar los datos: {e}')
270
271     except Exception as e:
272         print(f'Error al borrar los datos: {e}')
273
274     except Exception as e:
275         print(f'Error al borrar los datos: {e}')
276
277     except Exception as e:
278         print(f'Error al borrar los datos: {e}')
279
280     except Exception as e:
281         print(f'Error al borrar los datos: {e}')
282
283     except Exception as e:
284         print(f'Error al borrar los datos: {e}')
285
286     except Exception as e:
287         print(f'Error al borrar los datos: {e}')
288
289     except Exception as e:
290         print(f'Error al borrar los datos: {e}')
291
292     except Exception as e:
293         print(f'Error al borrar los datos: {e}')
294
295     except Exception as e:
296         print(f'Error al borrar los datos: {e}')
297
298     except Exception as e:
299         print(f'Error al borrar los datos: {e}')
300
301     except Exception as e:
302         print(f'Error al borrar los datos: {e}')
303
304     except Exception as e:
305         print(f'Error al borrar los datos: {e}')
306
307     except Exception as e:
308         print(f'Error al borrar los datos: {e}')
309
310     except Exception as e:
311         print(f'Error al borrar los datos: {e}')
312
313     except Exception as e:
314         print(f'Error al borrar los datos: {e}')
315
316     except Exception as e:
317         print(f'Error al borrar los datos: {e}')
318
319     except Exception as e:
320         print(f'Error al borrar los datos: {e}')
321
322     except Exception as e:
323         print(f'Error al borrar los datos: {e}')
324
325     except Exception as e:
326         print(f'Error al borrar los datos: {e}')
327
328     except Exception as e:
329         print(f'Error al borrar los datos: {e}')
330
331     except Exception as e:
332         print(f'Error al borrar los datos: {e}')
333
334     except Exception as e:
335         print(f'Error al borrar los datos: {e}')
336
337     except Exception as e:
338         print(f'Error al borrar los datos: {e}')
339
340     except Exception as e:
341         print(f'Error al borrar los datos: {e}')
342
343     except Exception as e:
344         print(f'Error al borrar los datos: {e}')
345
346     except Exception as e:
347         print(f'Error al borrar los datos: {e}')
348
349     except Exception as e:
350         print(f'Error al borrar los datos: {e}')
351
352     except Exception as e:
353         print(f'Error al borrar los datos: {e}')
354
355     except Exception as e:
356         print(f'Error al borrar los datos: {e}')
357
358     except Exception as e:
359         print(f'Error al borrar los datos: {e}')
360
361     except Exception as e:
362         print(f'Error al borrar los datos: {e}')
363
364     except Exception as e:
365         print(f'Error al borrar los datos: {e}')
366
367     except Exception as e:
368         print(f'Error al borrar los datos: {e}')
369
370     except Exception as e:
371         print(f'Error al borrar los datos: {e}')
372
373     except Exception as e:
374         print(f'Error al borrar los datos: {e}')
375
376     except Exception as e:
377         print(f'Error al borrar los datos: {e}')
378
379     except Exception as e:
380         print(f'Error al borrar los datos: {e}')
381
382     except Exception as e:
383         print(f'Error al borrar los datos: {e}')
384
385     except Exception as e:
386         print(f'Error al borrar los datos: {e}')
387
388     except Exception as e:
389         print(f'Error al borrar los datos: {e}')
390
391     except Exception as e:
392         print(f'Error al borrar los datos: {e}')
393
394     except Exception as e:
395         print(f'Error al borrar los datos: {e}')
396
397     except Exception as e:
398         print(f'Error al borrar los datos: {e}')
399
400     except Exception as e:
401         print(f'Error al borrar los datos: {e}')
402
403     except Exception as e:
404         print(f'Error al borrar los datos: {e}')
405
406     except Exception as e:
407         print(f'Error al borrar los datos: {e}')
408
409     except Exception as e:
410         print(f'Error al borrar los datos: {e}')
411
412     except Exception as e:
413         print(f'Error al borrar los datos: {e}')
414
415     except Exception as e:
416         print(f'Error al borrar los datos: {e}')
417
418     except Exception as e:
419         print(f'Error al borrar los datos: {e}')
420
421     except Exception as e:
422         print(f'Error al borrar los datos: {e}')
423
424     except Exception as e:
425         print(f'Error al borrar los datos: {e}')
426
427     except Exception as e:
428         print(f'Error al borrar los datos: {e}')
429
430     except Exception as e:
431         print(f'Error al borrar los datos: {e}')
432
433     except Exception as e:
434         print(f'Error al borrar los datos: {e}')
435
436     except Exception as e:
437         print(f'Error al borrar los datos: {e}')
438
439     except Exception as e:
440         print(f'Error al borrar los datos: {e}')
441
442     except Exception as e:
443         print(f'Error al borrar los datos: {e}')
444
445     except Exception as e:
446         print(f'Error al borrar los datos: {e}')
447
448     except Exception as e:
449         print(f'Error al borrar los datos: {e}')
450
451     except Exception as e:
452         print(f'Error al borrar los datos: {e}')
453
454     except Exception as e:
455         print(f'Error al borrar los datos: {e}')
456
457     except Exception as e:
458         print(f'Error al borrar los datos: {e}')
459
460     except Exception as e:
461         print(f'Error al borrar los datos: {e}')
462
463     except Exception as e:
464         print(f'Error al borrar los datos: {e}')
465
466     except Exception as e:
467         print(f'Error al borrar los datos: {e}')
468
469     except Exception as e:
470         print(f'Error al borrar los datos: {e}')
471
472     except Exception as e:
473         print(f'Error al borrar los datos: {e}')
474
475     except Exception as e:
476         print(f'Error al borrar los datos: {e}')
477
478     except Exception as e:
479         print(f'Error al borrar los datos: {e}')
480
481     except Exception as e:
482         print(f'Error al borrar los datos: {e}')
483
484     except Exception as e:
485         print(f'Error al borrar los datos: {e}')
486
487     except Exception as e:
488         print(f'Error al borrar los datos: {e}')
489
490     except Exception as e:
491         print(f'Error al borrar los datos: {e}')
492
493     except Exception as e:
494         print(f'Error al borrar los datos: {e}')
495
496     except Exception as e:
497         print(f'Error al borrar los datos: {e}')
498
499     except Exception as e:
500         print(f'Error al borrar los datos: {e}')
501
502     except Exception as e:
503         print(f'Error al borrar los datos: {e}')
504
505     except Exception as e:
506         print(f'Error al borrar los datos: {e}')
507
508     except Exception as e:
509         print(f'Error al borrar los datos: {e}')
510
511     except Exception as e:
512         print(f'Error al borrar los datos: {e}')
513
514     except Exception as e:
515         print(f'Error al borrar los datos: {e}')
516
517     except Exception as e:
518         print(f'Error al borrar los datos: {e}')
519
520     except Exception as e:
521         print(f'Error al borrar los datos: {e}')
522
523     except Exception as e:
524         print(f'Error al borrar los datos: {e}')
525
526     except Exception as e:
527         print(f'Error al borrar los datos: {e}')
528
529     except Exception as e:
530         print(f'Error al borrar los datos: {e}')
531
532     except Exception as e:
533         print(f'Error al borrar los datos: {e}')
534
535     except Exception as e:
536         print(f'Error al borrar los datos: {e}')
537
538     except Exception as e:
539         print(f'Error al borrar los datos: {e}')
540
541     except Exception as e:
542         print(f'Error al borrar los datos: {e}')
543
544     except Exception as e:
545         print(f'Error al borrar los datos: {e}')
546
547     except Exception as e:
548         print(f'Error al borrar los datos: {e}')
549
550     except Exception as e:
551         print(f'Error al borrar los datos: {e}')
552
553     except Exception as e:
554         print(f'Error al borrar los datos: {e}')
555
556     except Exception as e:
557         print(f'Error al borrar los datos: {e}')
558
559     except Exception as e:
560         print(f'Error al borrar los datos: {e}')
561
562     except Exception as e:
563         print(f'Error al borrar los datos: {e}')
564
565     except Exception as e:
566         print(f'Error al borrar los datos: {e}')
567
568     except Exception as e:
569         print(f'Error al borrar los datos: {e}')
570
571     except Exception as e:
572         print(f'Error al borrar los datos: {e}')
573
574     except Exception as e:
575         print(f'Error al borrar los datos: {e}')
576
577     except Exception as e:
578         print(f'Error al borrar los datos: {e}')
579
580     except Exception as e:
581         print(f'Error al borrar los datos: {e}')
582
583     except Exception as e:
584         print(f'Error al borrar los datos: {e}')
585
586     except Exception as e:
587         print(f'Error al borrar los datos: {e}')
588
589     except Exception as e:
590         print(f'Error al borrar los datos: {e}')
591
592     except Exception as e:
593         print(f'Error al borrar los datos: {e}')
594
595     except Exception as e:
596         print(f'Error al borrar los datos: {e}')
597
598     except Exception as e:
599         print(f'Error al borrar los datos: {e}')
600
601     except Exception as e:
602         print(f'Error al borrar los datos: {e}')
603
604     except Exception as e:
605         print(f'Error al borrar los datos: {e}')
606
607     except Exception as e:
608         print(f'Error al borrar los datos: {e}')
609
610     except Exception as e:
611         print(f'Error al borrar los datos: {e}')
612
613     except Exception as e:
614         print(f'Error al borrar los datos: {e}')
615
616     except Exception as e:
617         print(f'Error al borrar los datos: {e}')
618
619     except Exception as e:
620         print(f'Error al borrar los datos: {e}')
621
622     except Exception as e:
623         print(f'Error al borrar los datos: {e}')
624
625     except Exception as e:
626         print(f'Error al borrar los datos: {e}')
627
628     except Exception as e:
629         print(f'Error al borrar los datos: {e}')
630
631     except Exception as e:
632         print(f'Error al borrar los datos: {e}')
633
634     except Exception as e:
635         print(f'Error al borrar los datos: {e}')
636
637     except Exception as e:
638         print(f'Error al borrar los datos: {e}')
639
640     except Exception as e:
641         print(f'Error al borrar los datos: {e}')
642
643     except Exception as e:
644         print(f'Error al borrar los datos: {e}')
645
646     except Exception as e:
647         print(f'Error al borrar los datos: {e}')
648
649     except Exception as e:
650         print(f'Error al borrar los datos: {e}')
651
652     except Exception as e:
653         print(f'Error al borrar los datos: {e}')
654
655     except Exception as e:
656         print(f'Error al borrar los datos: {e}')
657
658     except Exception as e:
659         print(f'Error al borrar los datos: {e}')
660
661     except Exception as e:
662         print(f'Error al borrar los datos: {e}')
663
664     except Exception as e:
665         print(f'Error al borrar los datos: {e}')
666
667     except Exception as e:
668         print(f'Error al borrar los datos: {e}')
669
670     except Exception as e:
671         print(f'Error al borrar los datos: {e}')
672
673     except Exception as e:
674         print(f'Error al borrar los datos: {e}')
675
676     except Exception as e:
677         print(f'Error al borrar los datos: {e}')
678
679     except Exception as e:
680         print(f'Error al borrar los datos: {e}')
681
682     except Exception as e:
683         print(f'Error al borrar los datos: {e}')
684
685     except Exception as e:
686         print(f'Error al borrar los datos: {e}')
687
688     except Exception as e:
689         print(f'Error al borrar los datos: {e}')
690
691     except Exception as e:
692         print(f'Error al borrar los datos: {e}')
693
694     except Exception as e:
695         print(f'Error al borrar los datos: {e}')
696
697     except Exception as e:
698         print(f'Error al borrar los datos: {e}')
699
700     except Exception as e:
701         print(f'Error al borrar los datos: {e}')
702
703     except Exception as e:
704         print(f'Error al borrar los datos: {e}')
705
706     except Exception as e:
707         print(f'Error al borrar los datos: {e}')
708
709     except Exception as e:
710         print(f'Error al borrar los datos: {e}')
711
712     except Exception as e:
713         print(f'Error al borrar los datos: {e}')
714
715     except Exception as e:
716         print(f'Error al borrar los datos: {e}')
717
718     except Exception as e:
719         print(f'Error al borrar los datos: {e}')
720
721     except Exception as e:
722         print(f'Error al borrar los datos: {e}')
723
724     except Exception as e:
725         print(f'Error al borrar los datos: {e}')
726
727     except Exception as e:
728         print(f'Error al borrar los datos: {e}')
729
730     except Exception as e:
731         print(f'Error al borrar los datos: {e}')
732
733     except Exception as e:
734         print(f'Error al borrar los datos: {e}')
735
736     except Exception as e:
737         print(f'Error al borrar los datos: {e}')
738
739     except Exception as e:
740         print(f'Error al borrar los datos: {e}')
741
742     except Exception as e:
743         print(f'Error al borrar los datos: {e}')
744
745     except Exception as e:
746         print(f'Error al borrar los datos: {e}')
747
748     except Exception as e:
749         print(f'Error al borrar los datos: {e}')
750
751     except Exception as e:
752         print(f'Error al borrar los datos: {e}')
753
754     except Exception as e:
755         print(f'Error al borrar los datos: {e}')
756
757     except Exception as e:
758         print(f'Error al borrar los datos: {e}')
759
760     except Exception as e:
761         print(f'Error al borrar los datos: {e}')
762
763     except Exception as e:
764         print(f'Error al borrar los datos: {e}')
765
766     except Exception as e:
767         print(f'Error al borrar los datos: {e}')
768
769     except Exception as e:
770         print(f'Error al borrar los datos: {e}')
771
772     except Exception as e:
773         print(f'Error al borrar los datos: {e}')
774
775     except Exception as e:
776         print(f'Error al borrar los datos: {e}')
777
778     except Exception as e:
779         print(f'Error al borrar los datos: {e}')
780
781     except Exception as e:
782         print(f'Error al borrar los datos: {e}')
783
784     except Exception as e:
785         print(f'Error al borrar los datos: {e}')
786
787     except Exception as e:
788         print(f'Error al borrar los datos: {e}')
789
790     except Exception as e:
791         print(f'Error al borrar los datos: {e}')
792
793     except Exception as e:
794         print(f'Error al borrar los datos: {e}')
795
796     except Exception as e:
797         print(f'Error al borrar los datos: {e}')
798
799     except Exception as e:
800         print(f'Error al borrar los datos: {e}')
801
802     except Exception as e:
803         print(f'Error al borrar los datos: {e}')
804
805     except Exception as e:
806         print(f'Error al borrar los datos: {e}')
807
808     except Exception as e:
809         print(f'Error al borrar los datos: {e}')
810
811     except Exception as e:
812         print(f'Error al borrar los datos: {e}')
813
814     except Exception as e:
815         print(f'Error al borrar los datos: {e}')
816
817     except Exception as e:
818         print(f'Error al borrar los datos: {e}')
819
820     except Exception as e:
821         print(f'Error al borrar los datos: {e}')
822
823     except Exception as e:
824         print(f'Error al borrar los datos: {e}')
825
826     except Exception as e:
827         print(f'Error al borrar los datos: {e}')
828
829     except Exception as e:
830         print(f'Error al borrar los datos: {e}')
831
832     except Exception as e:
833         print(f'Error al borrar los datos: {e}')
834
835     except Exception as e:
836         print(f'Error al borrar los datos: {e}')
837
838     except Exception as e:
839         print(f'Error al borrar los datos: {e}')
840
841     except Exception as e:
842         print(f'Error al borrar los datos: {e}')
843
844     except Exception as e:
845         print(f'Error al borrar los datos: {e}')
846
847     except Exception as e:
848         print(f'Error al borrar los datos: {e}')
849
850     except Exception as e:
851         print(f'Error al borrar los datos: {e}')
852
853     except Exception as e:
854         print(f'Error al borrar los datos: {e}')
855
856     except Exception as e:
857         print(f'Error al borrar los datos: {e}')
858
859     except Exception as e:
860         print(f'Error al borrar los datos: {e}')
861
862     except Exception as e:
863         print(f'Error al borrar los datos: {e}')
864
865     except Exception as e:
866         print(f'Error al borrar los datos: {e}')
867
868     except Exception as e:
869         print(f'Error al borrar los datos: {e}')
870
871     except Exception as e:
872         print(f'Error al borrar los datos: {e}')
873
874     except Exception as e:
875         print(f'Error al borrar los datos: {e}')
876
877     except Exception as e:
878         print(f'Error al borrar los datos: {e}')
879
880     except Exception as e:
881         print(f'Error al borrar los datos: {e}')
882
883     except Exception as e:
884         print(f'Error al borrar los datos: {e}')
885
886     except Exception as e:
887         print(f'Error al borrar los datos: {e}')
888
889     except Exception as e:
890         print(f'Error al borrar los datos: {e}')
891
892     except Exception as e:
893         print(f'Error al borrar los datos: {e}')
894
895     except Exception as e:
896         print(f'Error al borrar los datos: {e}')
897
898     except Exception as e:
899         print(f'Error al borrar los datos: {e}')
900
901     except Exception as e:
902         print(f'Error al borrar los datos: {e}')
903
904     except Exception as e:
905         print(f'Error al borrar los datos: {e}')
906
907     except Exception as e:
908         print(f'Error al borrar los datos: {e}')
909
910     except Exception as e:
911         print(f'Error al borrar los datos: {e}')
912
913     except Exception as e:
914         print(f'Error al borrar los datos: {e}')
915
916     except Exception as e:
917         print(f'Error al borrar los datos: {e}')
918
919     except Exception as e:
920         print(f'Error al borrar los datos: {e}')
921
922     except Exception as e:
923         print(f'Error al borrar los datos: {e}')
924
925     except Exception as e:
926         print(f'Error al borrar los datos: {e}')
927
928     except Exception as e:
929         print(f'Error al borrar los datos: {e}')
929

```

Figura 5: Código realizado en python

```

162         elif opcion_usuario == '4':
163             print('Saliedo al menú inicial...')
164             break
165         else:
166             print('Opción no válida. Por favor, elija una opción entre 1 y 4.')
167
168     elif respuesta == 'N':
169         # Registrar nuevo usuario
170         nombre = input('Ingrese su nombre: ').strip()
171         carne = input('Ingrese su carne (9 dígitos): ').strip()
172
173         if contiene_caracteres_especiales(nombre):
174             print('El nombre no debe contener caracteres especiales.')
175             continue
176
177         if not carne.isdigit() or len(carne) != 9:
178             print('El carné debe contener exactamente 9 dígitos numéricos.')
179             continue
180
181         cursor.execute('INSERT INTO usuario (nombre, carne) VALUES (%s, %s);', (nombre,
182 carne))
183
184         conn.commit()
185         print('Usuario registrado correctamente. Ahora puedes iniciar sesión.')
186     else:
187         print('Respuesta no válida. Debes ingresar "Y" o "N".')
188
189 # Cerrar la conexión a la base de datos
190 cursor.close()
191 conn.close()
192
193

```

Figura 6: Código realizado en python

2. Octave

```

1 if (exist('OCTAVE_VERSION', 'builtin') == 0)
2     % Estamos en Octave
3     pkg load database; % Cargar el paquete para interactuar con bases de datos
4 end
5
6 % Inicialización de conexión con la base de datos
7 conn = pg_connect(sprintf('dbname=%s;user=%s;host=%s;port=%s;password=%s;',
8 'postgres', 'postgres', 'localhost', '5432', 'postgres', 'postgres'));
9
10 % Nombre del archivo de texto para guardar los resultados
11 archivo_texto = 'verificaciones.txt';
12
13 % Verificar si el archivo de texto existe, si no, crearlo
14 if exist(archivo_texto, 'file') == 0
15     fid = fopen(archivo_texto, 'w');
16     fclose(fid);
17 else
18     disp('El archivo de texto ya existe.');
```

Figura 7: Código realizado en octave

```

42     if isempty(result.data)
43         disp('Usuario no encontrado. Por favor, registre primero los datos del usuario.');
```

Figura 8: Código realizado en octave

```

81     case 1
82         % Verificar que ya hay un usuario registrado
83         nombre = nombre;
84         carne = carne;
85
86         % Buscar al usuario en la base de datos
87         result = pg_exec_params(conn, 'SELECT id FROM usuario WHERE nombre = $1 AND carne = $2;',
88             (nombre, carne));
89
90         if isempty(result.data)
91             disp('Usuario no encontrado. Por favor, registre los datos del usuario primero.');
```

Figura 9: Código realizado en octave

```

121         if es_primo
122             disp('El número es primo.');
```

```

123         else
124             disp('El número no es un número primo.');
```

```

125         end
126     end
127
128     % Verificar si el número es perfecto
129     suma_divisores = 0;
130     for i = 1:(numero - 1)
131         if mod(numero, i) == 0
132             suma_divisores = suma_divisores + i;
133         end
134     end
135
136     if suma_divisores == numero
137         es_perfecto = true;
138         disp('El número es un número perfecto.');
```

```

139     else
140         es_perfecto = false;
141         disp('El número no es un número perfecto.');
```

```

142     end
143
144     % Verificar si el número es palindromo
145     num_str = num2str(numero); % Convertir el número a cadena de texto
146     if strcmp(num_str, flip(num_str))
147         es_palindromo = true;
148         disp('El número es un número palindromo.');
```

```

149     else
150         es_palindromo = false;
151         disp('El número no es un número palindromo.');
```

```

152     end
153
154     % Insertar los resultados en la base de datos
155     pq_exec_params(conn, ['INSERT INTO verificaciones (id_usuario, numero,
156     es_primo, es_perfecto, es_palindromo) ...
157     VALUES ($1, $2, $3, $4, $5)'], ...
158     {id_usuario, numero, es_primo, es_perfecto,
159     es_palindromo});
158     disp('Los resultados han sido guardados en la base de datos.');
```

Figura 10: Código realizado en octave

B. Diagrama de flujo

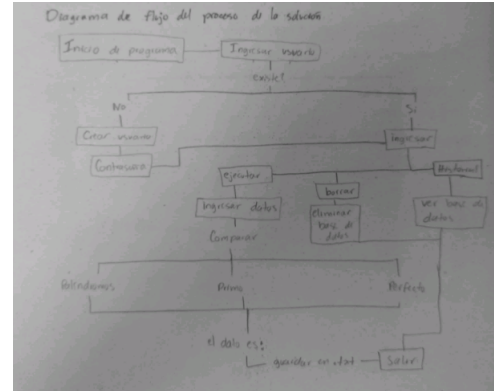


Figura 13: Diagrama de flujo del programa realizado

C. Algoritmo

```

159     % Guardar los resultados en el archivo de texto
160     fid = fopen(archivo_texto, 'a'); % Abrir el archivo en modo "append"
161     fprintf(fid, 'Usuario: %s (Carné: %s)\n', nombre, carne);
162     fprintf(fid, 'Número: %d\n', numero);
163     fprintf(fid, 'Es primo: %d\n', es_primo);
164     fprintf(fid, 'Es perfecto: %d\n', es_perfecto);
165     fprintf(fid, 'Es palindromo: %d\n', es_palindromo);
166     fclose(fid); % Cerrar el archivo
167     disp('Los resultados también han sido guardados en el archivo de texto.');
```

```

168     end
169
170     end
171
172     % Mostrar el historial de datos directamente desde la tabla 'verificaciones'
173     result = pq_exec_params(conn, 'SELECT id_usuario, numero, es_primo, es_perfecto,
174     es_palindromo FROM verificaciones');
```

```

175     if isempty(result.data)
176         disp('No hay datos registrados.');
```

```

177     else
178         disp('----- Historial de verificaciones -----');
```

```

179         disp(['ID Usuario\tNumero\tPrimo\tPerfecto\tPalindromo']);
180
181         % Iterar sobre los resultados y mostrarlos de forma legible
182         for i = 1:size(result.data, 1)
183             id_usuario = result.data(i, 1);
184             numero = result.data(i, 2);
185             es_primo = result.data(i, 3);
186             es_perfecto = result.data(i, 4);
187             es_palindromo = result.data(i, 5);
188
189             % Mostrar la información de manera formateada
190             fprintf('%d\t%d\t%d\t%d\t%d\n', id_usuario, numero, es_primo, es_perfecto,
191             es_palindromo);
192         end
193     end
194
195     % Borrar todos los datos de la tabla 'verificaciones'
196     try
197         pq_exec_params(conn, 'DELETE FROM verificaciones');
```

```

198         disp('Todos los datos de la tabla de verificaciones han sido borrados.');
```

```

199     catch
200     end
201
202     % Borrar los datos del archivo de texto 'verificacion.txt'
203     if exist('verificacion.txt', 'file') == 2
204         % Si el archivo existe, vaciarlo
205         fclose(fopen('verificacion.txt', 'w'));
206         disp('Los datos del archivo de texto "verificacion.txt" han sido borrados.');
```

```

207     else
208         disp('El archivo "verificacion.txt" no existe.');
```

```

209     end
210
211     catch
212         disp('Error al borrar los datos.');
```

```

213     end
214
215     case 4
216         disp('Saliendo del menú principal...');
```

```

217     break;
218
219     otherwise
220         disp('Opción no válida. Intente de nuevo.');
```

```

221     end
222
223     else
224         disp('Opción no válida. Intente de nuevo.');
```

```

225     end
226
227     end
228
229     end
230
231     end
232
233     end
234
235     end
236
237     end
238
239     end
240
241     end
242
243     end
244
245     end
246
247     end
248
249     end
250
251     end
252
253     end
254
255     end
256
257     end
258
259     end
260
261     end
262
263     end
264
265     end
266
267     end
268
269     end
270
271     end
272
273     end
274
275     end
276
277     end
278
279     end
280
281     end
282
283     end
284
285     end
286
287     end
288
289     end
290
291     end
292
293     end
294
295     end
296
297     end
298
299     end
300
301     end
302
303     end
304
305     end
306
307     end
308
309     end
310
311     end
312
313     end
314
315     end
316
317     end
318
319     end
320
321     end
322
323     end
324
325     end
326
327     end
328
329     end
330
331     end
332
333     end
334
335     end
336
337     end
338
339     end
340
341     end
342
343     end
344
345     end
346
347     end
348
349     end
350
351     end
352
353     end
354
355     end
356
357     end
358
359     end
360
361     end
362
363     end
364
365     end
366
367     end
368
369     end
370
371     end
372
373     end
374
375     end
376
377     end
378
379     end
380
381     end
382
383     end
384
385     end
386
387     end
388
389     end
390
391     end
392
393     end
394
395     end
396
397     end
398
399     end
400
401     end
402
403     end
404
405     end
406
407     end
408
409     end
410
411     end
412
413     end
414
415     end
416
417     end
418
419     end
420
421     end
422
423     end
424
425     end
426
427     end
428
429     end
430
431     end
432
433     end
434
435     end
436
437     end
438
439     end
440
441     end
442
443     end
444
445     end
446
447     end
448
449     end
450
451     end
452
453     end
454
455     end
456
457     end
458
459     end
460
461     end
462
463     end
464
465     end
466
467     end
468
469     end
470
471     end
472
473     end
474
475     end
476
477     end
478
479     end
480
481     end
482
483     end
484
485     end
486
487     end
488
489     end
490
491     end
492
493     end
494
495     end
496
497     end
498
499     end
500
501     end
502
503     end
504
505     end
506
507     end
508
509     end
510
511     end
512
513     end
514
515     end
516
517     end
518
519     end
520
521     end
522
523     end
524
525     end
526
527     end
528
529     end
530
531     end
532
533     end
534
535     end
536
537     end
538
539     end
540
541     end
542
543     end
544
545     end
546
547     end
548
549     end
550
551     end
552
553     end
554
555     end
556
557     end
558
559     end
560
561     end
562
563     end
564
565     end
566
567     end
568
569     end
570
571     end
572
573     end
574
575     end
576
577     end
578
579     end
580
581     end
582
583     end
584
585     end
586
587     end
588
589     end
590
591     end
592
593     end
594
595     end
596
597     end
598
599     end
600
601     end
602
603     end
604
605     end
606
607     end
608
609     end
610
611     end
612
613     end
614
615     end
616
617     end
618
619     end
620
621     end
622
623     end
624
625     end
626
627     end
628
629     end
630
631     end
632
633     end
634
635     end
636
637     end
638
639     end
640
641     end
642
643     end
644
645     end
646
647     end
648
649     end
650
651     end
652
653     end
654
655     end
656
657     end
658
659     end
660
661     end
662
663     end
664
665     end
666
667     end
668
669     end
670
671     end
672
673     end
674
675     end
676
677     end
678
679     end
680
681     end
682
683     end
684
685     end
686
687     end
688
689     end
690
691     end
692
693     end
694
695     end
696
697     end
698
699     end
700
701     end
702
703     end
704
705     end
706
707     end
708
709     end
710
711     end
712
713     end
714
715     end
716
717     end
718
719     end
720
721     end
722
723     end
724
725     end
726
727     end
728
729     end
730
731     end
732
733     end
734
735     end
736
737     end
738
739     end
740
741     end
742
743     end
744
745     end
746
747     end
748
749     end
750
751     end
752
753     end
754
755     end
756
757     end
758
759     end
760
761     end
762
763     end
764
765     end
766
767     end
768
769     end
770
771     end
772
773     end
774
775     end
776
777     end
778
779     end
780
781     end
782
783     end
784
785     end
786
787     end
788
789     end
790
791     end
792
793     end
794
795     end
796
797     end
798
799     end
800
801     end
802
803     end
804
805     end
806
807     end
808
809     end
810
811     end
812
813     end
814
815     end
816
817     end
818
819     end
820
821     end
822
823     end
824
825     end
826
827     end
828
829     end
830
831     end
832
833     end
834
835     end
836
837     end
838
839     end
840
841     end
842
843     end
844
845     end
846
847     end
848
849     end
850
851     end
852
853     end
854
855     end
856
857     end
858
859     end
860
861     end
862
863     end
864
865     end
866
867     end
868
869     end
870
871     end
872
873     end
874
875     end
876
877     end
878
879     end
880
881     end
882
883     end
884
885     end
886
887     end
888
889     end
890
891     end
892
893     end
894
895     end
896
897     end
898
899     end
900
901     end
902
903     end
904
905     end
906
907     end
908
909     end
910
911     end
912
913     end
914
915     end
916
917     end
918
919     end
920
921     end
922
923     end
924
925     end
926
927     end
928
929     end
930
931     end
932
933     end
934
935     end
936
937     end
938
939     end
940
941     end
942
943     end
944
945     end
946
947     end
948
949     end
950
951     end
952
953     end
954
955     end
956
957     end
958
959     end
960
961     end
962
963     end
964
965     end
966
967     end
968
969     end
970
971     end
972
973     end
974
975     end
976
977     end
978
979     end
980
981     end
982
983     end
984
985     end
986
987     end
988
989     end
990
991     end
992
993     end
994
995     end
996
997     end
998
999     end
1000
1001     end
1002
1003     end
1004
1005     end
1006
1007     end
1008
1009     end
1010
1011     end
1012
1013     end
1014
1015     end
1016
1017     end
1018
1019     end
1020
1021     end
1022
1023     end
1024
1025     end
1026
1027     end
1028
1029     end
1030
1031     end
1032
1033     end
1034
1035     end
1036
1037     end
1038
1039     end
1040
1041     end
1042
1043     end
1044
1045     end
1046
1047     end
1048
1049     end
1050
1051     end
1052
1053     end
1054
1055     end
1056
1057     end
1058
1059     end
1060
1061     end
1062
1063     end
1064
1065     end
1066
1067     end
1068
1069     end
1070
1071     end
1072
1073     end
1074
1075     end
1076
1077     end
1078
1079     end
1080
1081     end
1082
1083     end
1084
1085     end
1086
1087     end
1088
1089     end
1090
1091     end
1092
1093     end
1094
1095     end
1096
1097     end
1098
1099     end
1100
1101     end
1102
1103     end
1104
1105     end
1106
1107     end
1108
1109     end
1110
1111     end
1112
1113     end
1114
1115     end
1116
1117     end
1118
1119     end
1120
1121     end
1122
1123     end
1124
1125     end
1126
1127     end
1128
1129     end
1130
1131     end
1132
1133     end
1134
1135     end
1136
1137     end
1138
1139     end
1140
1141     end
1142
1143     end
1144
1145     end
1146
1147     end
1148
1149     end
1150
1151     end
1152
1153     end
1154
1155     end
1156
1157     end
1158
1159     end
1160
1161     end
1162
1163     end
1164
1165     end
1166
1167     end
1168
1169     end
1170
1171     end
1172
1173     end
1174
1175     end
1176
1177     end
1178
1179     end
1180
1181     end
1182
1183     end
1184
1185     end
1186
1187     end
1188
1189     end
1190
1191     end
1192
1193     end
1194
1195     end
1196
1197     end
1198
1199     end
1200
1201     end
1202
1203     end
1204
1205     end
1206
1207     end
1208
1209     end
1210
1211     end
1212
1213     end
1214
1215     end
1216
1217     end
1218
1219     end
1220
1221     end
1222
1223     end
1224
1225     end
1226
1227     end
1228
1229     end
1230
1231     end
1232
1233     end
1234
1235     end
1236
1237     end
1238
1239     end
1240
1241     end
1242
1243     end
1244
1245     end
1246
1247     end
1248
1249     end
1250
1251     end
1252
1253     end
1254
1255     end
1256
1257     end
1258
1259     end
1260
1261     end
1262
1263     end
1264
1265     end
1266
1267     end
1268
1269     end
1270
1271     end
1272
1273     end
1274
1275     end
1276
1277     end
1278
1279     end
1280
1281     end
1282
1283     end
1284
1285     end
1286
1287     end
1288
1289     end
1290
1291     end
1292
1293     end
1294
1295     end
1296
1297     end
1298
1299     end
1300
1301     end
1302
1303     end
1304
1305     end
1306
1307     end
1308
1309     end
1310
1311     end
1312
1313     end
1314
1315     end
1316
1317     end
1318
1319     end
1320
1321     end
1322
1323     end
1324
1325     end
1326
1327     end
1328
1329     end
1330
1331     end
1332
1333     end
1334
1335     end
1336
1337     end
1338
1339     end
1340
1341     end
1342
1343     end
1344
1345     end
1346
1347     end
1348
1349     end
1350
1351     end
1352
1353     end
1354
1355     end
1356
1357     end
1358
1359     end
1360
1361     end
1362
1363     end
1364
1365     end
1366
1367     end
1368
1369     end
1370
1371     end
1372
1373     end
1374
1375     end
1376
1377     end
1378
1379     end
1380
1381     end
1382
1383     end
1384
1385     end
1386
1387     end
1388
1389     end
1390
1391     end
1392
1393     end
1394
1395     end
1396
1397     end
1398
1399     end
1400
1401     end
1402
1403     end
1404
1405     end
1406
1407     end
1408
1409     end
1410
1411     end
1412
1413     end
1414
1415     end
1416
1417     end
1418
1419     end
1420
1421     end
1422
1423     end
1424
1425     end
1426
1427     end
1428
1429     end
1430
1431     end
1432
1433     end
1434
1435     end
1436
1437     end
1438
1439     end
1440
1441     end
1442
1443     end
1444
1445     end
1446
1447     end
1448
1449     end
1450
1451     end
1452
1453     end
1454
1455     end
1456
1457     end
1458
1459     end
1460
1461     end
1462
1463     end
1464
1465     end
1466
1467     end
1468
1469     end
1470
1471     end
1472
1473     end
1474
1475     end
1476
1477     end
1478
1479     end
1480
1481     end
1482
1483     end
1484
1485     end
1486
1487     end
1488
1489     end
1490
1491     end
1492
1493     end
1494
1495     end
1496
1497     end
1498
1499     end
1500
1501     end
1502
1503     end
1504
1505     end
1506
1507     end
1508
1509     end
1510
1511     end
1512
1513     end
1514
1515     end
1516
1517     end
1518
1519     end
1520
1521     end
1522
1523     end
1524
1525     end
1526
1527     end
1528
1529     end
1530
1531     end
1532
1533     end
1534
1535     end
1536
1537     end
1538
1539     end
1540
1541     end
1542
1543     end
1544
1545     end
1546
1547     end
1548
1549     end
1550
1551     end
1552
1553     end
1554
1555     end
1556
1557     end
1558
1559     end
1560
1561     end
1562
1563     end
1564
1565     end
1566
1567     end
1568
1569     end
1570
1571     end
1572
1573     end
1574
1575     end
1576
1577     end
1578
1579     end
1580
1581     end
1582
1583     end
1584
1585     end
1586
1587     end
1588
1589     end
1590
1591     end
1592
1593     end
1594
1595     end
1596
1597     end
1598
1599     end
1600
1601     end
1602
1603     end
1604
1605     end
1606
1607     end
1608
1609     end
1610
1611     end
1612
1613     end
1614
1615     end
1616
1617     end
1618
1619     end
1620
1621     end
1622
1623     end
1624
1625     end
1626
1627     end
1628
1629     end
1630
1631     end
1632
1633     end
1634
1635     end
1636
1637     end
1638
1639     end
1640
1641     end
1642
1643     end
1644
1645     end
1646
1647     end
1648
1649     end
1650
1651     end
1652
1653     end
1654
1655     end
1656
1657     end
1658
1659     end
1660
1661     end
1662
1663     end
1664
1665     end
1666
1667     end
1668
1669     end
1670
1671     end
1672
1673     end
1674
1675     end
1676
1677     end
1678
1679     end
1680
1681     end
1682
1683     end
1684
1685     end
1686
1687     end
1688
1689     end
1690
1691     end
1692
1693     end
1694
1695     end
1696
1697     end
1698
1699     end
1700
1701     end
1702
1703     end
1704
1705     end
1706
1707     end
1708
1709     end
1710
1711     end
1712
1713     end
1714
1715     end
1716
1717     end
1718
1719     end
1720
1721     end
1722
1723     end
1724
1725     end
1726
1727     end
1728
1729     end
1730
1731     end
1732
1733     end
1734
1735     end
1736
1737     end
1738
1739     end
1740
1741     end
1742
1743     end
1744
1745     end
1746
1747     end
1748
1749     end
1750
1751     end
1752
1753     end
1754
1755     end
1756
1757     end
1758
1759     end
1760
1761     end
1762
1763     end
1764
1765     end
1766
1767     end
1768
1769     end
1770
1771     end
1772
1773     end
1774
1775     end
1776
1777     end
1778
1779     end
1780
1781     end
1782
1783     end
1784
1785     end
1786
1787     end
1788
1789     end
1790
1791     end
1792
1793     end
1794
1795     end
1796
1797     end
1798
1799     end
1800
1801     end
1802
1803     end
1804
1805     end
1806
1807     end
1808
1809     end
1810
1811     end
1812
1813     end
1814
1815     end
1816
1817     end
1818
1819     end
1820
1821     end
1822
1823     end
1824
1825     end
1826
1827     end
1828
1829     end
1830
1831     end
1832
1833     end
1834
1835     end
1836
1837     end
1838
1839     end
1840
1841     end
1842
1843     end
1844
1845     end
1846
1847     end
1848
1849     end
1850
1851     end
1852
1853     end
1854
1855     end
1856
1857     end
1858
1859     end
1860
1861     end
1862
1863     end
1864
1865     end
1866
1867     end
1868
1869     end
1870
1871     end
1872
1873     end
1874
1875     end
1876
1877     end
1878
1879     end
1880
1881     end
1882
1883     end
1884
1885     end
1886
1887     end
1888
1889     end
1890
1891     end
1892
1893     end
1894
1895     end
1896
1897     end
1898
1899     end
1900
1901     end
1902
1903     end
1904
1905     end
1906
1907     end
1908
1909     end
1910
1911     end
1912
1913     end
1914
1915     end
1916
1917     end
1918
1919     end
1920
1921     end
1922
1923     end
1924
1925     end
1926
1927     end
1928
1929     end
1930
1931     end
1932
1933     end
1934
1935     end
1936
1937     end
1938
1939     end
1940
1941     end
1942
1943     end
1944
1945     end
1946
1947     end
1948
1949     end
1950
1951     end
1952
1953     end
1954
1955     end
1956
1957     end
1958
1959     end
1960
1961     end
1962
1963     end
1964
1965     end
1966
1967     end
1968
1969     end
1970
1971     end
1972
1973     end
1974
1975     end
1976
1977     end
1978
1979     end
1980
1981     end
1982
1983     end
1984
1985     end
1986
1987     end
1988
1989     end
1990
1991     end
1992
1993     end
1994
1995     end
1996
1997     end
1998
1999     end
2000
2001     end
2002
2003     end
2004
2005     end
2006
2007     end
2008
2009     end
2010
2011     end
2012
2013     end
2014
2015     end
2016
2017     end
2018
2019     end
2020
2021     end
2022
2023     end
2024
2025     end
2026
2027     end
2028
2029     end
2030
2031     end
2032
2033     end
2034
2035     end
2036
2037     end
2038
2039     end
2040
2041     end
2042
2043     end
2044
2045     end
2046
2047     end
2048
2049     end
2050
2051     end
2052
2053     end
2054
2055     end
2056
2057     end
2058
2059     end
2060
2061     end
2062
2063     end
2064
2065     end
2066
2067     end
2068
2069     end
2070
2071     end
2072
2073     end
2074
2075     end
2076
2077     end
2078
2079     end
2080
2081     end
2082
2083     end
2084
2085     end
2086
2087     end
2088
2089     end
2090
2091     end
2092
2093     end
2094
2095     end
2096
2097     end
2098
2099     end
2100
2101     end
210
```

```
--- Menú Inicial ---
1. Ingresar Usuario
2. Salir
Ingrese su elección: 1
¿Ya tienes un usuario registrado? (Y/N): Y
Ingrese su nombre: Boris
Ingrese su carné (9 dígitos): 199900023
¡Bienvenido de nuevo!
--- Menú de Usuario ---
1. Ingresar datos
2. Ver historial de datos
3. Borrar datos
4. Salir
Ingrese su elección: |
```

Figura 17: Funcionamiento del programa en Python

```
Ingrese su elección: 1
Tipos de números
1. Un número primo es aquel que solo tiene dos divisores: 1 y el mismo número.
2. Un número perfecto es aquel que es igual a la suma de sus divisores propios (exceptuando el mismo número).
3. Un número palíndromo es aquel que se lee igual de izquierda a derecha que de derecha a izquierda.
Ingrese el dato que desea verificar (solo números): 33
El número no es un número primo.
El número no es un número perfecto.
El número es un número palíndromo.
Los resultados han sido guardados en la base de datos.
Los resultados también han sido guardados en el archivo de texto.
--- Menú Principal ---
1. Ingresar datos
2. Ver historial de datos
3. Borrar datos
4. Salir
Ingrese su elección: 2
----- Historial de verificaciones -----
ID Usuario\Número\Primo\Perfecto\Palíndromo
1 33 0 0 1
--- Menú Principal ---
1. Ingresar datos
2. Ver historial de datos
3. Borrar datos
4. Salir
Ingrese su elección: |
```

Figura 22: Funcionamiento del programa en Octave

```
Ingrese su elección: 1
Ingrese el número a verificar: 513
El número no es primo.
El número no es perfecto.
El número no es palíndromo.
Los resultados han sido guardados en la base de datos.
Datos guardados correctamente.
--- Menú de Usuario ---
1. Ingresar datos
2. Ver historial de datos
3. Borrar datos
4. Salir
Ingrese su elección: |
```

Figura 18: Funcionamiento del programa en Python

```
Ingrese su elección: 3
Todos los datos de la tabla de verificaciones han sido borrados.
Los datos del archivo de texto "verificacion.txt" han sido borrados.
--- Menú Principal ---
1. Ingresar datos
2. Ver historial de datos
3. Borrar datos
4. Salir
Ingrese su elección: 4
Saliendo del menú principal...
--- Menú Inicial ---
1. Ingresar Usuario
2. Salir
Ingrese su elección: |
```

Figura 23: Funcionamiento del programa en Octave

```
Ingrese su elección: 2
----- Historial de verificaciones -----
ID Usuario  Número  Primo  Perfecto  Palíndromo
7           513     False  False     False
--- Menú de Usuario ---
```

Figura 19: Funcionamiento del programa en Python

3. Base de datos

```
--- Menú de Usuario ---
1. Ingresar datos
2. Ver historial de datos
3. Borrar datos
4. Salir
Ingrese su elección: 3
Todos los datos de la tabla de verificaciones han sido borrados.
Los datos del archivo de texto "resultados_verificaciones.txt" han sido borrados.
--- Menú de Usuario ---
```

Figura 20: Funcionamiento del programa en Python

2. Octave

```
>> tarea
El archivo de texto ya existe.
--- Menú Inicial ---
1. Ingresar Usuario
2. Salir
Ingrese su elección: 1
¿Ya tienes un usuario registrado? (Y/N): N
Ingrese su nombre: B@ris
Ingrese su carné: 123
El nombre solo debe contener letras.
--- Menú Inicial ---
1. Ingresar Usuario
2. Salir
Ingrese su elección: 1
¿Ya tienes un usuario registrado? (Y/N): Y
Ingrese su nombre: Alan
Ingrese su carné: 202100023
¡Bienvenido de nuevo!
--- Menú Principal ---
1. Ingresar datos
2. Ver historial de datos
3. Borrar datos
4. Salir
Ingrese su elección: |
```

Figura 21: Funcionamiento del programa en Octave

id	nombre	carne
1	Alan	202100023
2	Alan	202100023
3	Mafer	123456789
4	Sara	202012345
5	Luisa	098765432
6	Marcos	234567890
7	Boris	199900023

Figura 24: Base de datos con usuarios registrados

id	id_usuario	numero	es_primo	es_perfecto	es_palindromo
1	25	7	1441	false	false
2	26	7	6	false	true
3	27	7	737	false	false

Figura 25: Base de datos con datos verificados

4. Archivos de texto

```

Usuario: Boris (Carné: 199900023)
Número: 31
Es primo: 1
Es perfecto: 0
Es palíndromo: 0

Usuario: Boris (Carné: 199900023)
Número: 6
Es primo: 0
Es perfecto: 1
Es palíndromo: 1

Usuario: Boris (Carné: 199900023)
Número: 2920
Es primo: 0
Es perfecto: 0
Es palíndromo: 0

Usuario: Boris (Carné: 199900023)
Número: 141
Es primo: 0
Es perfecto: 0
Es palíndromo: 1

```

Figura 26: Datos registrados en archivo de texto octave

```

Usuario: Alan (Carné: 202100023)
Número: 3
Es primo: 1
Es perfecto: 0
Es palíndromo: 1

Usuario: Alan (Carné: 202100023)
Número: 6
Es primo: 0
Es perfecto: 1
Es palíndromo: 1

Usuario: Alan (Carné: 202100023)
Número: 141
Es primo: 0
Es perfecto: 0
Es palíndromo: 1

```

Figura 27: Datos registrados en archivo de texto python

E. Repositorio privado

Repositorio creado en Github

F. Videos del funcionamiento

Video del funcionamiento del programa

III. CONCLUSIONES

- * La implementación de las soluciones en Python y Octave demuestra la adaptabilidad de los algoritmos matemáticos y de gestión de datos en distintos lenguajes de programación. Python aporta un enfoque moderno y versátil, mientras que Octave se destaca por sus capacidades en cálculos matemáticos y científicos.
- * En Python, la integración con PostgreSQL muestra cómo interactuar con bases de datos de manera eficiente, garantizando la persistencia de los datos y la gestión estructurada de la información de los usuarios. En Octave, aunque más limitado en cuanto a bases de datos, se demostró cómo procesar datos y realizar verificaciones matemáticas.
- * Tanto en Python como en Octave, se desarrollaron funciones para verificar si un número es primo, perfecto o palíndromo. Esto subraya la importancia de adaptar algoritmos matemáticos a diferentes entornos de programación, destacando las fortalezas de Octave en operaciones numéricas y la flexibilidad de Python.
- * El programa en Python no solo se conecta a la base de datos, sino que también gestiona archivos de texto como respaldo, lo que proporciona una solución híbrida para la persistencia de datos. En Octave, la lógica de almacenamiento fue más enfocada a la demostración y procesamiento de datos sin conexión con bases de datos.
- * La inclusión de validaciones de entrada tanto en Python como en Octave asegura la robustez y seguridad de los programas, evitando errores comunes y mejorando la experiencia del usuario.