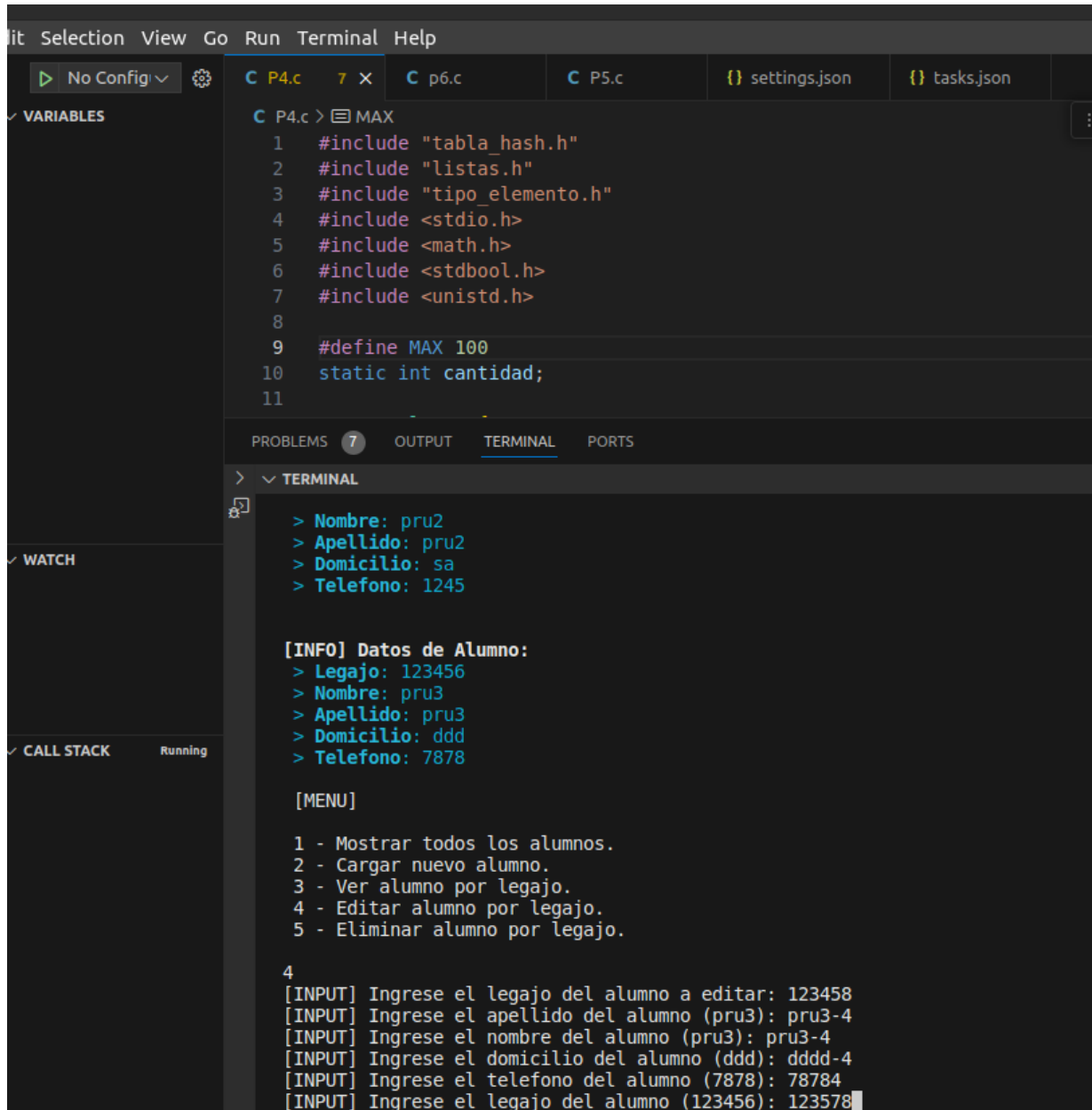


RESULTADO DE LA CORRECCIÓN: **APROBADO-**

En el ejercicio 4 puedo modificar el legajo y no debería. No muestra la tabla hash con lo cuál no se puede ver cómo se va conformando.



```

C P4.c > MAX
1  #include "tabla_hash.h"
2  #include "listas.h"
3  #include "tipo_elemento.h"
4  #include <stdio.h>
5  #include <math.h>
6  #include <stdbool.h>
7  #include <unistd.h>
8
9  #define MAX 100
10 static int cantidad;
11
PROBLEMS 7 OUTPUT TERMINAL PORTS
> Nombre: pru2
> Apellido: pru2
> Domicilio: sa
> Telefono: 1245

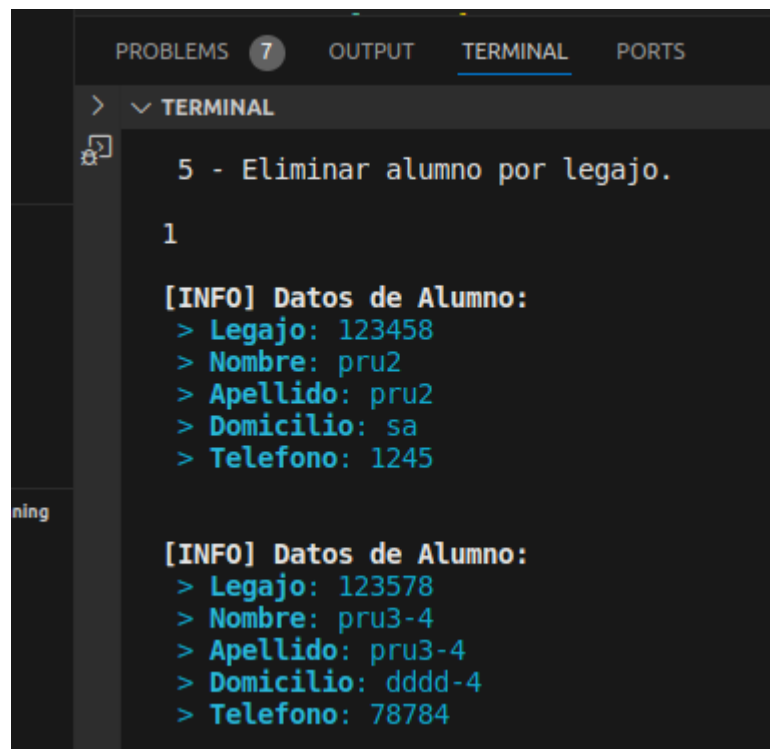
[INFO] Datos de Alumno:
> Legajo: 123456
> Nombre: pru3
> Apellido: pru3
> Domicilio: ddd
> Telefono: 7878

[MENU]

1 - Mostrar todos los alumnos.
2 - Cargar nuevo alumno.
3 - Ver alumno por legajo.
4 - Editar alumno por legajo.
5 - Eliminar alumno por legajo.

4
[INPUT] Ingrese el legajo del alumno a editar: 123458
[INPUT] Ingrese el apellido del alumno (pru3): pru3-4
[INPUT] Ingrese el nombre del alumno (pru3): pru3-4
[INPUT] Ingrese el domicilio del alumno (ddd): dddd-4
[INPUT] Ingrese el telefono del alumno (7878): 78784
[INPUT] Ingrese el legajo del alumno (123456): 123578
```

GRUPO 19 – Correcciones Trabajo Práctico: TABLAS HASH

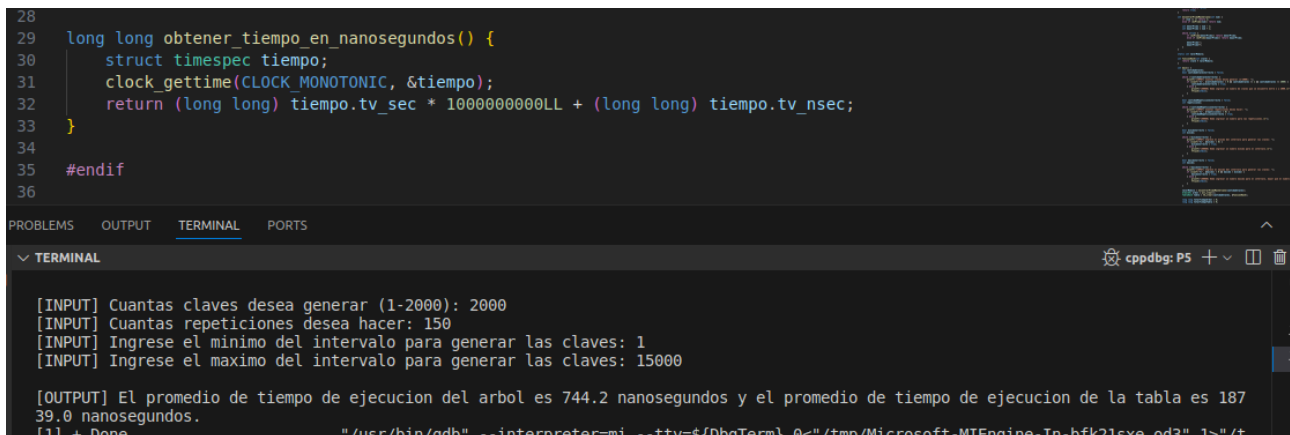


```
PROBLEMS 7 OUTPUT TERMINAL PORTS
> v TERMINAL
5 - Eliminar alumno por legajo.
1

[INFO] Datos de Alumno:
> Legajo: 123458
> Nombre: pru2
> Apellido: pru2
> Domicilio: sa
> Telefono: 1245

[INFO] Datos de Alumno:
> Legajo: 123578
> Nombre: pru3-4
> Apellido: pru3-4
> Domicilio: dddd-4
> Telefono: 78784
```

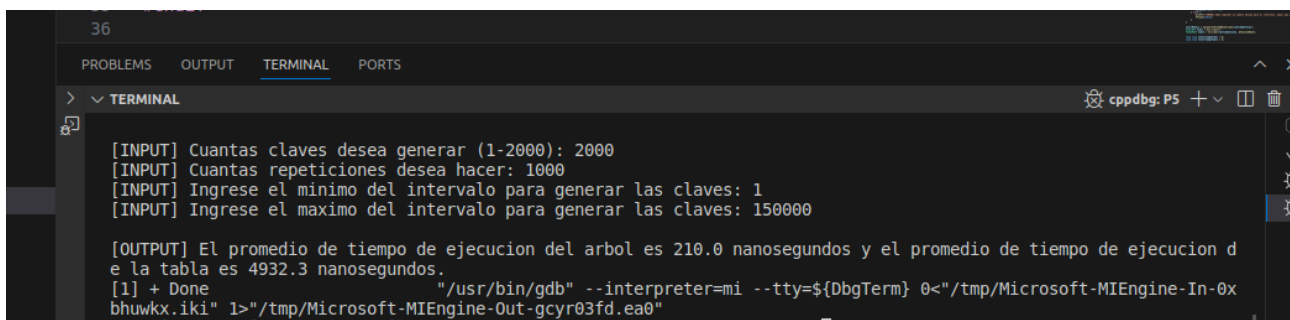
En el ejercicio 5 no hay conclusión. Solamente muestra los tiempos de acceso a cada estructura.



```
28
29 long long obtener_tiempo_en_nanosegundos() {
30     struct timespec tiempo;
31     clock_gettime(CLOCK_MONOTONIC, &tiempo);
32     return (long long) tiempo.tv_sec * 1000000000LL + (long long) tiempo.tv_nsec;
33 }
34
35 #endif
36
```

```
PROBLEMS OUTPUT TERMINAL PORTS
v TERMINAL
[INPUT] Cuantas claves desea generar (1-2000): 2000
[INPUT] Cuantas repeticiones desea hacer: 150
[INPUT] Ingrese el minimo del intervalo para generar las claves: 1
[INPUT] Ingrese el maximo del intervalo para generar las claves: 15000

[OUTPUT] El promedio de tiempo de ejecucion del arbol es 744.2 nanosegundos y el promedio de tiempo de ejecucion de la tabla es 18739.0 nanosegundos.
[1] + Done
```



```
36
PROBLEMS OUTPUT TERMINAL PORTS
> v TERMINAL
[INPUT] Cuantas claves desea generar (1-2000): 2000
[INPUT] Cuantas repeticiones desea hacer: 1000
[INPUT] Ingrese el minimo del intervalo para generar las claves: 1
[INPUT] Ingrese el maximo del intervalo para generar las claves: 150000

[OUTPUT] El promedio de tiempo de ejecucion del arbol es 210.0 nanosegundos y el promedio de tiempo de ejecucion d
e la tabla es 4932.3 nanosegundos.
[1] + Done
```