

---

**Ejercitación – Serie G**  
**Valores aleatorios y variables static**

---

Realizar las funciones solicitadas en un archivo .c, y el main de la aplicación en otro archivo .c

---

Ejercicio 1: (el dado)

Implemente una función que genere y retorne números aleatorios del 1 al 6.

```
int dado (void);
```

Ejercicio 2: (cubilete)

Implementar una función que imprima en pantalla la resultante de haber lanzado 5 dados.

Ejercicio 3: (cubilete')

Modificar la función anterior de manera de poder realizar una serie de lanzamientos e indicar en cada lanzamiento que dados volver a arrojar y cuáles no.

La indicación de que dados volver a lanzar se debe realizar con una variable de 8bits, a la cual se le debe asignar un bit a cada dado (total 5 bits).

El bit en 1 (uno) significa que el dado no debe ser lanzado, y en 0 (cero) que sí.

Si el parámetro es cero (todos sus bits en cero) significa que se deben arrojar los 5 dados.

La función debe devolver la cantidad de dados arrojados.

Ejercicio 4: (cubilete'')

Agregar a la función anterior la posibilidad de devolver el resultado de la tirada (es decir el valor de todos los dados) en una sola variable.

**Nota:** Se debe asignar una cantidad de bits y posición dentro de una variable a cada dado. La variable debe ser del tamaño adecuado para albergar los 5 dados.

Ejercicio 5: (jugada)

Realizar que pasándole el valor devuelto por la función del ejercicio 4 (resultado de la tirada), imprima y retorne si se trata de una generala, un poker, full o escalera.

Ejercicio 6: (jugada numérica)

Realizar que pasándole el valor devuelto por la función del ejercicio 4 (resultado de la tirada) y un valor de dado a evaluar, retorne la cantidad de veces que ese dado aparece en la tirada (valor de 0 a 5).

Ejercicio 7: (Solitario)

Realizar una aplicación para jugar a la generala en modalidad obligada (aquella que primero se hace el 1, luego el 2 y así hasta el 6, pasando posteriormente a la escalera, full, poker, generala y en algunos casos generala doble).

Luego de cada turno o jugada se debe imprimir el puntaje obtenido y el acumulado.

Ejercicio 8: (partida)

Modificar la aplicación para permitir jugar a 2 personas en simultáneo.

Finalizado el juego la aplicación debe indicar al participante ganador.

Ejercicio 9: para pensar

Que modificaciones debería considerar para una aplicación en donde los participantes decidan luego de haber realizado los tiros correspondientes a un turno, elija la jugada a anotar.

Que modificaciones debería hacer para permitir una cantidad variable de participantes. Por ejemplo 10 o 3.

Como se podría hacer un torneo.

Ejercicio 10: (punto flotante)

Realizar una aplicación que permita generar un número aleatorio de punto flotante de simple precisión acotado en rango.

---