

Algoritmos y Programación I (95.11) – Curso Santisi – 2^{do} parcialito – 20/10/2023

Resolver los siguientes problemas en forma clara y legible en código ISO-C99.

1. Implementar una función `bool misma_cantidad(uint32_t n)`; que devuelva `true` si la cantidad de bits prendidos en las posiciones impares de `n` es la misma que la de bits prendidos en las posiciones pares.

Por ejemplo, `misma_cantidad(0100 1011) -> true` porque están prendidos los bits 0, 1, 3 y 6; y dos de ellos están en posiciones pares y dos en impares.

2. Se quiere implementar una función `char *join(char *cadenas[], size_t n, char separador)`; que reciba un arreglo de `n` cadenas de caracteres y devuelva una única cadena que contenga a todas ellas con el `separador` intercalado.

Por ejemplo, `join({"Hola", "que", "tal"}, 3, ' ') -> "Hola que tal"`.

3. Se tiene una estructura que representa a una carta de baraja española. La misma contiene el valor de la carta, que es un número entre 1 y 12, y el palo, que es un enumerativo con las etiquetas `ORO`, `ESPADA`, `BASTO` y `COPA`.

- Declarar el enumerativo `enum palo` y la estructura `struct carta`.
- Redefinir `struct carta` como `carta_t`.
- Teniendo ya implementada una función `carta_t *sacar_carta_del_mazo()`; que al ser llamada devuelve una nueva carta implementar la función:

```
carta_t **tirar_reyes(size_t *n);
```

que llame a `sacar_carta_del_mazo()` la cantidad de veces que haga falta hasta obtener un rey (valor 12). Debe devolver un arreglo con cada una de las cartas sacadas hasta el momento (incluyendo el rey). Se debe devolver, además, la cantidad de cartas sacadas en `n`.

¡Suerte! :)