

# Laboratório 3: Mapa de bits

*Entrega até segunda, 1/4, às 6h*

Nesse laboratório, vamos realizar a implementação de representação de conjuntos com número de elementos limitado, implementados em **set.h** e **set.c**.

As operações bit a bit de C permitem que manipulemos bits individuais de um inteiro sem sinal (unsigned).

Podemos utilizar esses operadores para implementar, através de um array de bits, representações de conjuntos com número de elementos limitados. Podemos representar um conjunto por um array de char ou int contendo um número suficiente de bits para que cada bit indique se o elemento correspondente está ou não no conjunto.

Vamos considerar conjuntos de inteiros de 0 a 31. Neste caso, um único valor unsigned int é suficiente para a representação desses conjuntos.

1. Implemente as funções da interface set.h. O arquivo set.c contém um esboço de implementação. A função set\_show, que mostra os elementos de um conjunto, já está implementada, para facilitar seus testes, e a função set\_is\_member também. Use o arquivo test.c como base para seus testes.

Crie suas funções uma a uma e vá testando-as à medida que são criadas.

Funções para implementar:

- `void set_insert(Set *set, int i);`
- `void set_remove(Set *set, int i);`
- `Set* set_copy(Set *set);`
- `Set *set_union(Set *set1, Set *set2);`
- `Set *set_intersection(Set *set1, Set *set2);`
- `Set *set_difference(Set *set1, Set *set2);`
- `int set_is_subset(Set *set1, Set *set2);`
- `int set_is_equal(Set *set1, Set *set2);`
- `int set_number_of_elements(Set *set);`
- `Set *set_complement(Set *set);`

Faça upload dos arquivos **set.c** no EAD até dia 1 de abril, segunda, às 6h. Lembre-se de fazer a entrega mesmo que não tenha chegado ao final do exercício.