

Bitを活用しよう！

2017-08-08 競プロ強化カリキュラムLT会

@mt_caret

二分木の全探索

2択を1ビットで表し $[0, 2^N)$ の範囲でforループで回すと全ての状態を列挙することができる。

参考: [実践・最強最速のアルゴリズム勉強会 第二回講義資料](#)

例 (ABC002-D 派閥)

- N 人の間にある M 個の人間関係を与えられ、全員が全員の間と関係を持っている最大人数のグループの人数を計算せよ
- $1 \leq N \leq 12$
- $0 \leq M \leq N(N - 1)/2$

例（ABC002-D 派閥）

1. 全てのグループのとり方を列挙
2. グループ全員が全員の間と関係を持っているかどうかを走査
3. 条件を満たすようなグループの人数の最大人数を見つける

例 (ABC002-D 派閥)

~省略~

```
int ans = -1
for (int r = 0; r < (1 << 12); r++) {
    bool flag = true;
    for (int i = 0; i < 11; i++) {
        if ((r >> i) % 2) {
            for (int j = i + 1; j < 12; j++) {
                if ((r >> j) % 2) flag = flag && table[i][j];
            }
        }
    }
    if (flag) ans = max(ans, __builtin_popcount(r));
}
cout << ans << endl;
```

~省略~

- $(1 \ll n) \ 2^N$
- $(a \gg b) \% 2$ aのbビット目が立っているか
- `__builtin_popcount(x)` xの立っているビット数

小手先のテクニック色々

- スワップ `if (a != b) { a ^= b; b ^= a; a ^= b; }`
`std::swap()` 使いましょうという話もある
- 最下位ビットを0にする `a &= a-1`
- 立っているビットの偶奇 `__builtin_parity(x)`

参考

- 明日使えないすごいビット演算
- へ、変態っ！！読めないからやめてっ！bit使ったデータ構造・アルゴリズム実装集
- 6.59 Other Built-in Functions Provided by GCC

発展的なトピック色々

- BitDP → 蟻本や[AtCoder資料](#)参照
- FFT用の高速なBit反転
- [Bitwise Tricks & Techniques](#)