

# 位运算题目练习

---

## 人员

---

左子毅、朱奕鸣、杨洋、刘子淇、赵清航、周子航、于迦浩、刘佳赫 到课

## 作业检查

---

左子毅 完成

朱奕鸣 未完成

杨洋 完成

刘子淇 未完成

赵清航 未完成

周子航 完成

于迦浩 完成

刘佳赫 第一次进班

## 作业

---

<https://www.luogu.com.cn/contest/171699>

A、B、C、D 题

## 课堂表现

---

课堂纪律整体不错，希望同学们以后可以继续保持

不过同学们整体代码水平有待提高，需要多刷题题单，提高自己代码能力

## 课堂内容

---

Maximum And Queries (easy version)

从高位往低位判断：

判断能否把这n个数的对应这一位都变为1

如果n个数这一位都能变为1，就变；

否则，看下一位

```
#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int maxn = 1e5 + 5;
int w[maxn], a[maxn];

void solve(int n) {
    for (int i = 1; i <= n; ++i) w[i] = a[i];

    LL m, ans = 0; cin >> m;
    for (int i = 62; i >= 0; --i) {
        LL sum = 0, limit = (1LL<<i);
        for (int j = 1; j <= n; ++j) {
            if (w[j] & limit) continue;
            sum += limit - w[j]%limit;
            if (sum > m) break;
        }
        if (sum > m) continue;

        m -= sum;
        ans += limit;
        for (int j = 1; j <= n; ++j) {
            if (w[j] & limit) continue;
            w[j] = limit;
        }
    }

    cout << ans << endl;
}

int main()
{
    int n, T; cin >> n >> T;
    for (int i = 1; i <= n; ++i) cin >> a[i];
    while (T -- ) solve(n);
    return 0;
}
```

从高位往低位判断，同时要保留之前check的结果

dp:

$f[i][j]$ : 代表前 $i$ 个数，能否分成 $j$ 组

```
#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int N = 60;
LL w[N], p[N];
bool f[N][N];

LL sum(int l, int r) { return p[r] - p[l-1]; }

bool check(int n, int m, LL res) {
    memset(f, false, sizeof(f));
    f[0][0] = true;
    for (int i = 1; i <= n; ++i) {
        for (int j = 1; j <= m; ++j) {
            for (int i2 = 0; i2 <= i-1; ++i2) {
                if (f[i2][j-1] && (sum(i2+1,i)&res)==res) {
                    f[i][j] = true;
                }
            }
        }
    }
    return f[n][m];
}

int main()
{
    int n, m; cin >> n >> m;
    for (int i = 1; i <= n; ++i) cin >> w[i], p[i] = p[i-1] + w[i];

    LL res = 0;
    for (int i = N; i >= 0; --i) {
        if (check(n, m, res|(1LL<<i))) {
            res |= (1LL<<i);
        }
    }
    cout << res << endl;
    return 0;
}
```

```
f[i] = (i-1) * (f[i-1] + f[i-2])
```

```
#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int maxn = 20 + 5;
LL f[maxn];

int main()
{
    int n; cin >> n;
    f[1] = 0, f[2] = 1;
    for (int i = 3; i <= n; ++i) {
        f[i] = (i-1) * (f[i-1] + f[i-2]);
    }
    cout << f[n] << endl;
    return 0;
}
```