

1 语言基础

1.1 C++ Snippets

1.1.1 define class

```
class Foo {
public:
    Foo() { this->a = 1; }
    int a;
private:
};
```

1.1.2 io

<iostream>

```
ostream &operator<<(ostream &o, Foo &f);
istream &operator>>(istream &i, Foo &f);
```

<cstdio>

```
freopen("data.in", "r", stdin);
freopen("data.out", "w", stdout);
fclose(stdin);
fclose(stdout);
```

1.1.3 other utilities

- <utility>: pair
- <bitset>

1.1.4 useful snippets

- mid of $[begin, end]$: $begin + (end - begin) / 2$
- $float = float$: $\text{fabs}(a - b) < 1e - 6$

2 DP

背包类. w : 背包大小, w_n : n 的重量, v_n : n 的价值,

```
// 最大价值
f(/* 0.. */ n, w) = max(
    /* 全选 */ f(n - 1, w - w_n) + v_n,
    /* 部分 */ ...,
    /* 不选 */ f(n - 1, w)
)
```

3 数学

3.1 位运算

- $n \&= n - 1$: 敲除最右边的 1 (Brian Kernighan)

3.2 数论

3.2.1 最大公约数 GCD

$gcd(a, b)$ (c++: <numeric>gcd)

```
func gcd(int a, b) int {
    if b == 1 {
        return a
    }
    return gcd(b, a % b)
}
```

```
// ax + by = gcd(a, b)
func exgcd(int a, b) (int x, y) {
    if b == 0 {
        x = 1; y = 0;
        return
    }
    y, x = exgcd(b, a % b)
    y -= x * (a / b)
    return
}
```

3.2.2 最小公倍数 LCM

$lcm(a, b)$ (c++: <numeric>lcm)

3.2.3 逆元

$a\%p$ 的逆元 x : $ax\%p = 1$

- $x, _ = exgcd(a, p)$
- 若 p 为素数: $x = a^{(p-2)}\%p$ (费马小定理)

应用场景

- $(n/a) \% p = nx \% p$

3.3 组合数学

3.3.1 排列 Permutation

$P(n, k) = n! / (n - k)!$

3.3.2 组合 Combination

```
C(n, k) = n! / (n - k)! / k!
        = C(n - 1, k - 1) + C(n - 1, k)
        = C(n - 1, k) * n / (n - k)
C(n, 0) = C(n, n) = 1
```

4 Platform

4.1 acm.ecnu.edu.cn

- `std::cin.eof()` not work?