Algorithm Library

mental2008

October 31, 2018

Contents

1 图论

2 String

3 黑科技

3.1 输入输出外挂

3.1.1 简单快读

```
// 整数 (int, long long)
template<typename T>
void Read(T &r){
        char c;
        while(c=getchar()){
                if(isdigit(c)){
                        r=c^0x30;break;
                }
        }
        while(isdigit(c=getchar()))
                r=r*10+(c^0x30);
}
```

3.1.2 真・快速读入

```
#include<cstdio>
//BUAA 输入挂, 可读 __int128
namespace FastIO {
        #define BUF_SIZE 10000000 //缓冲区大小可修改
        bool IOError = 0; //IOError == false 时表示处理到文件结
        ↪ 尾
        inline char NextChar() {
                static char buf[BUF_SIZE], *p1 = buf +
                ↪ BUF_SIZE, *pend = buf + BUF_SIZE;
                if(p1 == pend) {
                        p1 = buf;
                        pend = buf + fread(buf, 1, BUF_SIZE,
                        ↪ stdin);
                        if(pend == p1) {
                                IOError = 1;
                                return -1;
                        }
                }
                return *p1++;
        }
        inline bool Blank(char c) {
                return c == ' ' || c == '\n' || c == '\r' || c
                ↪ == '\t';
        }
```

```
        template<class T> inline void Read(T &x) {
                char c;
                while(Blank(c = NextChar()));
                if(!IOError) {
                        for(x = 0; '0' <= c && c <= '9'; c =
                        ↪  NextChar())
                                x = (x << 3) + (x << 1) + c -
                                ↪  '0';
                }
        }
}

/*
读入时候这样写：

int x;
FastIO::Read(x);

若要处理到文件末尾可以这样写：

while(FastIO::Read(x), FastIO::IOError == 0);
*/
```