

常用库函数

- `math.h`

```
double fabs(double x) 求浮点数x的绝对值
double round(double x)求浮点数x四舍五入后的值
double ceil(double x) 对浮点数x向上取整 double floor(double x) 对浮点数向下取整
double sin/cos/tan(double x) 求弧度制下浮点数x的三角函数值
double asin/acos/atan(double x)求浮点数x的反三角函数值（弧度）
double pow(double x,double y)求x的y次方
double log/log2/log10(double x) 求x在底数时e,2,10时的对数函数值
double sqrt(double x)求x的算术平方根
```

- `ctype.h`

```
int isalpha/isdigit/islower/isupper(char c) 判断字符c是否是字母，数字，小写字母，大写字母，如果是返回值为1，否则为0
char toupper/tolower(char c)将字符c转换为大写/小写字母
```

- `stdlib.h`

```
int abs(int x) 求int类型x的绝对值
long long labs(long long x) 求long long类型x的绝对值
```

字符串函数{ strcpy strcat拼接 strcmp(可以比较字典序) strncmp(a,b,num)比较字符串前num位
strncpy(b,a,n)copy a的前n位到b strset把字符串每一位设置成某字符 strnset前n位 strrchr(a,'c')在串中查找某字符最后一次出现的地址 strrev 倒转字符串 strspn(str1,str2) 该函数返回 str1 中第一个不在字符串 str2 中出现的字符下标。(返回值为int型) strstr查找字符串在字符串中第一次出现的地址strupr 将字符串全转换为大写 }

字符画打印

```
/*字符画打印， 将需要打印的字符画放入同目录的source.txt文件中， 打印输出至同目录target.txt文件
*/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
char n1[50] = "#include <stdio.h>\nint main(){\nprintf(", n2[5] = "\", n3[5] = "\\n\\n\"", n4[50] = "\\");\nreturn 0;\n}";
void WriteChar(FILE *fp, char a) {
    fputc(a, fp);
}
void ReadChar(FILE *fpr, FILE *fpw) {
```

```
char ch;
fputs(n2, fpw);
while((ch = fgetc(fpr)) != EOF) {
    if(ch == '\\' || ch == '\'' || ch == '\"')
        fputc('\\', fpw);
    if(ch == '%')
        fputc('%', fpw);
    if(ch == '\n') {
        fputs(n3, fpw);
        fputc(ch, fpw);
        fputs(n2, fpw);
    } else
        fputc(ch, fpw);
}
}
int main() {
    FILE *fp1 = NULL, *fp2 = NULL;
    char buff[255];
    fp1 = fopen("source.txt", "r");
    fp2 = fopen("target.txt", "w");
    fputs(n1, fp2);
    ReadChar(fp1, fp2);
    fputs(n4, fp2);
    fclose(fp1);
    fclose(fp2);
    return 0;
}
```

输入

- 单组数据输入

```
//输入int类型整数
int x;
scanf("%d",&x);
//输入long long类型整数
long long x;
scanf("%lld",&x);
//输入double类型浮点数
double x;
scanf("%lf",&x);
//输入单个字符
char c;
scanf("%c",&c);
//或者
c = getchar();
//输入数组
int a[110];
for(int i = 1; i <= n; i++)
    scanf("%d",&a[i]);
//输入字符串
```

```
char c[110];
for(int i = 1; i <= n; i++)
scanf("%c", &c[i]);
//或者
scanf("%s", c); //遇到空格或者换行符就停止, 下标从0开始
//或者
gets(c); //遇到换行符停止, 下标从0开始
```

- 多组输入

```
int t;
scanf("%d", &t);
while(t--)
{
    //读入或者处理数据
}
```

- 不定组输入

```
while(scanf("%d", &n) != EOF) //ctrl z 结束输入, 这里输入一个正整数只是举例
{
    //读入或者处理数据
}
```

最大公约数和最小公倍数

```
int gcd(int x, int y) //根据题目可以换成long long
{
    if(x == 0)
        return y;
    return gcd(y % x, x);
}
int lcm(int x, int y)
{
    return x * y / gcd(x, y);
}
```

最大值和最小值

```
#define max(a, b) (((a) > (b)) ? (a) : (b))
#define min(a, b) (((a) < (b)) ? (a) : (b)) //写在头文件之后, 也可以自己写函数
```

判断素数

时间复杂度 $O(\sqrt{x})$

```
int isprime(int x)//如果是质数返回1, 否则返回0
{
    if(x <= 1)
        return 0;
    for(int i = 2;i * i <= x;i++)
    {
        if(x % i == 0)
            return 0;
    }
    return 1;
}
```

计算两点间距离

```
double dis(double x1,double y1,double x2,double y2)//计算两点(x1,y1),(x2,y2)之间的距离
{
    return sqrt((x1 - x2) * (x1 - x2) + (y1 - y2) * (y1 - y2));
}
```

前缀和数组

预处理复杂度 $O(n)$, 单组查询 $O(1)$

```
int a[100010],pre[100010];
void pre()
{
    for(int i = 1;i <= n;i++)
        pre[i] = pre[i - 1] + a[i];
}
int query(int l,int r)//计算l到r区间和
{
    return pre[r] - pre[l - 1];
}
```

差分数组

```
int a[100010],dif[100010];
void pre()
{
    for(int i = 1;i <= n;i++)
        dif[i] = a[i] - a[i - 1];
}
```

```
void modify(int l,int r,int x)//在区间l,r上加上x
{
    d[l] += x,d[r + 1] -= x;
}
void solve()//最终的a数组
{
    for(int i = 1;i <= n;i++)
        a[i] = a[i - 1] + dif[i];
}
```