

# 高精度(阶乘,加法,减法,乘法,高精乘低精,高精除低精,高精度比较)模板.

最近深受高精度的困扰，想用`python`又不是很熟练，就把高精度板子都整理了一遍了，除了高精除高精没写，不想写了，太懒了。 *ps* :还是`python`香。

便于以后使用和回顾。

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

using namespace std;

typedef long long ll;

const int N=5e6+100;

#define mst(a) memset(a,0,sizeof a)

char a[N],b[N];

struct num{

    int len,c[N];

}A,B,ans;

num fac(int n){    //高精度阶乘

    num a;

    int p[N];

    mst(p);

    memset(a.c,0,sizeof a.c);

    a.c[1]=1,a.len=1;

    for(int i=1;i<=n;i++){

        for(int j=1;j<=a.len;j++){

            a.c[j]=i*a.c[j]+p[j];p[j]=0;

            if(a.c[j]>9){

                p[j+1]=a.c[j]/10;

                a.c[j]%=10;

                if(j==a.len) a.len++;

            }

        }

    }

    return a;

}

num add(num a,num b){ //高精加

    num ans;

    memset(ans.c,0,sizeof ans.c);
```

```

int l=max(a.len,b.len),p=0;

ans.len=1;

for(int i=1;i<=l;i++){

    ans.c[i]=a.c[i]+b.c[i]+p;

    p=ans.c[i]/10;

    ans.c[i]%=10;

}

if(p) ans.c[++ans.len]=p;

return ans;

}

num multi(num a,num b){ //高精乘

    num ans;

    memset(ans.c,0,sizeof ans.c);

    int l=a.len+b.len-1,p=0;

    for(int i=1;i<=a.len;i++)

        for(int j=1;j<=b.len;j++)

            ans.c[i+j-1]+=a.c[i]*b.c[j];

    for(int i=1;i<=l;i++){

        ans.c[i+1]+=ans.c[i]/10;

        ans.c[i]%=10;

    }

    if(ans.c[l+1]) l++;

    ans.len=l;

    return ans;

}

num sub(num a,num b){ //高精减

    int f=0;

    if(a.len<b.len) swap(a,b),f=1;

    else if(a.len==b.len){

        for(int i=1;i<=a.len;i++)

            if(a.c[i]<b.c[i]){

                f=1;

                swap(a,b);

                break;

            }

    }

    for(int i=1;i<=a.len;i++){

```

```

        int x=a.c[i]-b.c[i];

        if(x<0) x+=10,a.c[i+1]--;

        a.c[i]=x;
    }

    if(f) printf("-");

    int j=a.len;

    while(!a.c[j]&& j>1) j--;

    a.len=j;

    return a;
}

void l_add(num &a,int x){ //高精乘低精

    int p=0;

    for(int i=1;i<=a.len;i++){

        a.c[i]=a.c[i]*x+p;

        p=a.c[i]/10;

        a.c[i]%=10;

    }

    while(p){

        a.c[++a.len]=p;

        p/=10;

        a.c[a.len]%=10;

    }

}

void div(num &a,int x){ //高精除低精度.(向下取整)

    int p=0;

    for(int i=a.len;i>=1;i--){

        p=p*10+a.c[i];

        a.c[i]=p/x;

        p%=x;

    }

    while(!a.c[a.len]) a.len--;

}

bool jg(num a,num b){ //高精度比较

    if(a.len!=b.len) return a.len>b.len;

    else for(int i=1;i<=a.len;i++)

        if(a.c[i]!=b.c[i])

            return a.c[i]>b.c[i];

```

```

}

int main() {

    /*

    int x;

    scanf("%s%d", a+1, &x);

    A.len=strlen(a+1);

    for(int i=1;i<=A.len;i++)

        A.c[i]=a[A.len+1-i]-'0';

    div(A, x);

    for(int i=A.len;i>=1;i--)

        printf("%d", A.c[i]);

    scanf("%s%s", a+1, b+1);

    A.len=strlen(a+1), B.len=strlen(b+1);

    for(int i=1;i<=B.len;i++)

        B.c[i]=b[B.len+1-i]-'0';

    ans=add(A, B);

    for(int i=ans.len;i>=1;i--)

        printf("%d", ans.c[i]);

    */

    return 0;

}

```