struct np{

short n;

short p;

};

member variable:

private:

string num;

bool negative = false;

np cal[100]={{0}};

public:

int len = 0;

member function:

private:

void check();

檢查進位，前面是零的問題，直接修改本身，不回傳值。

void print();

印出cal，用來檢查數字是否正確。

big\_int merge(big\_int q);

merge兩個BigInt，兩個BigInt都不改變，回傳計算後的BigInt。

public:

1. big\_int();

default construtor，預設為正數，長度為零。

1. big\_int(string& c);

constructor，傳入字串。

1. big\_int(const big\_int &b);

copy constructor。

1. ~big\_int();

destructor。

1. big\_int operator+(big\_int q);

operator overload，兩個BigInt 相加，自身不變，回傳一個BigInt。

1. big\_int operator-(big\_int q);

operator overload，兩個BigInt 相減，自身不變，回傳一個BigInt。

1. big\_int operator-();

取相反數，自身不變，回傳一個BigInt。

1. big\_int operator\*(big\_int q);

兩個BigInt相乘，自身不變，回傳一個BigInt積。

1. big\_int operator/(big\_int q);

自身除以傳入值，自身不變，回傳一個BigInt商。

1. big\_int operator/(int n);

BigInt除以int，自身不變，回傳一個BigInt商。

1. big\_int operator%(big\_int q);

自身對傳入值取餘數，自身不變，回傳一個BigInt餘數。

1. big\_int operator%(int n);

BigInt對傳入的int取餘數，自身不變，回傳一個BigInt餘數。

1. int operator[](int n);

[]中放入要提取的位數，回傳BigInt中該位數是一個int，自身不變。

1. bool operator>(big\_int q);

如果自身大於傳入值，回傳true，否則回傳false，兩個BigInt都不變。

1. bool operator<(big\_int q);

如果自身小於傳入值，回傳true，否則回傳false，兩個BigInt都不變。

1. bool operator==(big\_int q);

如果自身等於傳入值，回傳true，否則回傳false，兩個BigInt都不變。

1. bool isPrime();

如果該BigInt是質數，回傳true，合數或無法判別回傳false，自身不變。

1. void operator=(big\_int q);

將本身改變成傳入的BigInt，被傳入值不變，不回傳東西。

1. void operator=(string s);

將本身改變成傳入的String，被傳入值不變，不回傳東西。

1. big\_int abs();

對本身取絕對值回傳一個BigInt，本身不變。

1. big\_int square();

對本身取平方回傳一個BigInt，本身不變。

1. friend ostream& operator<<(ostream& out,const big\_int& q);

因為有overload cout，所以設成friend。

1. friend istream& operator>>(istream& in,const big\_int& q);

因為有overload cout，所以設成friend。

