



## مقدمه

هدف از این تمرین آشنایی با سربارگذاری عملگرها<sup>۱</sup> و استفاده از این قابلیت در حوزه اعداد بزرگ<sup>۲</sup> با ممیز شناور است. همانطور که می‌دانید اعداد از نوع double، بازه هرچند بزرگ اما محدودی را پشتیبانی می‌کنند. اعداد بزرگ قرار است که این محدودیت را رفع کنند و بتوانند اعداد با هر تعداد رقم را پشتیبانی کنند. اعداد بزرگ در حوزه‌هایی همچون رمزنگاری و کیهان‌شناسی کاربرد دارد.

## اعداد بزرگ

برای مدل کردن اعداد بزرگ کلاس BigInt را طوری تعریف کنید که هر instance از آن، قابلیت نگهداری یک عدد را داشته باشد سپس لازم است که موارد زیر را برای این کلاس پیاده‌سازی کنید:

تابع سازنده :

```
BigInt();
BigInt(string number);

Example :
BigInt("122365745237513.12543878176480324203472");
```

سازنده اول یک نمونه ایجاد می‌کند و مقدارش را برابر صفر قرار می‌دهد و سازنده دوم عدد n را که به صورت یک رشته دریافت کرده‌است به عنوان مقدار اولیه نمونه جدید قرار می‌دهد.

عملگر جمع :

با سربارگذاری عملگر + این امکان را فراهم کنید که بتوان دو عدد بزرگ را با هم جمع کرد.

```
BigInt a("1123123");
BigInt b("1.234");
BigInt c()
c = a + b; // c = 1123124.234
```

عملگر انتساب ( assignment ) :

عملگر = را طوری پیاده‌سازی کنید که عدد بزرگ سمت راست عملگر جایگزین عدد بزرگ سمت چپ عملگر شود.

<sup>1</sup> Operator Overloading

<sup>2</sup> Big Numbers

### عملگر منطقی :

عملگر منطقی == را به گونه‌ای پیاده‌سازی کنید که در صورتی که اعداد سمت راست و چپ عملگر با یکدیگر برابر بودند مقدار true و در غیر این صورت مقدار false بازگرداند.

```
BigNum bn1("234.123");
BigNum bn2("345.124");
BigNum bn3("234.123");
bn1 == bn2; //false
bn3 == bn1; //true
```

### خروجی و ورودی اعداد بزرگ :

برای چاپ یک عدد بزرگ عملگر << را برای ostream سربارگذاری کنید. هم‌چنین برای دریافت عدد بزرگ به عنوان ورودی نیز عملگر >> را برای istream سربارگذاری کنید. در نمایش عدد بزرگ به عنوان خروجی توجه کنید که نباید صفرهای سمت چپ قسمت صحیح و سمت راست قسمت اعشار را چاپ کنید.

```
BigNum bn1();
BigNum bn2("234.7657");
BigNum bn3("-0567.321");
BigNum bn4("9876.3400");
cin >> bn1;
cout << bn2; // Output : 234.7657
cout << bn3; // Output : -567.321
cout << bn4; // Output : 9876.34
```

### ورودی :

در خط اول عدد n آمده است که نشان‌دهنده‌ی تعداد اعداد بزرگ می‌باشد. در n خط بعدی به ترتیب همه‌ی اعداد بزرگ وارد می‌شوند. سپس عدد k می‌آید که نشان‌دهنده‌ی تعداد عملگرهاست. در k خط بعدی در هر خط به ترتیب شماره‌ی عدد اول (i)، عملگر == یا + و در نهایت شماره‌ی عدد دوم (j) را نشان می‌دهد.

$$1 \leq i \leq n, \quad 1 \leq j \leq n, \quad 1 \leq n \leq 1000, \quad 1 \leq k \leq 1000$$

### خروجی :

برنامه‌ی شما باید در k خط خروجی هر کدام از عملیات‌های محاسباتی را چاپ کند. (در صورتی که پاسخ یک عدد بزرگ بود مقدار آن عدد و در صورتی که پاسخ یک boolean بود true و false چاپ کند)

برای درک بهتر به ورودی و خروجی نمونه دقت کنید.

## ورودی و خروجی نمونه

ورودی	خروجی
3 1234651897953.13413432 45572.1232700 45572.12327 3 1 + 3 3 == 2 1 == 2	1234651943525.25740432 true false

## نحوه‌ی تحویل :

شما باید یک فایل با فرمت .cpp به اسم H8-SID در قسمت مربوط به هندزآن هشتم آپلود کنید. برای مثال اگر شماره‌ی دانشجویی شما 810195000 می‌باشد نام فایل آپلود شده باید H8-810195000 باشد. به فرمت ورودی ، خروجی و اسم فایل آپلود شده دقت فرمایید.