BigInt Library For C and C++

What is BigInt?

BigInt is a inbuilt library provided by java to handle very large values efficiently and effectively.So I have implemented same functionality in C because it doesn't support any such thing to deal with large values.

Problem:
Overflows and calculation headaches while dealing with large values

- Many a time in programming or problem solving , we have to deal with large values which can’t be stored inside any inbuilt data type.

- So it becomes very difficult to work in those scenario and we have to work under some modulo and can’t get exact answer.

- Difficulty while dealing with heavy computations on large values.

The solution (To Store the number in form of string)

- The first idea was to use char array provided by C language to store the number but then I realized that by doing this so I will again restrict myself to some number of digits.
- The second solution was to use string class provided by C++ but later I realized that C doesn’t support string class.

- Finally, I selected the second because strings are dynamic in nature so by using strings to represent the number, we have overcome the problem of limited number of digits.

Applications

1. To calculate factorial of bigger values.
2. To get the exact answer instead of under some modulo
3. No need to take care of overflows
4. To calculate exact value of x ^ y for any value of x and y.
5. To find total number of solutions for any particular NP - Hard problem.
6. Helps a lot in competitive programming where we need to deal with very large values.

DEMO

PROGRAM

vagrant@myvm18: ~

```
vagrant@myvm18:~$ cat main.c
#include <stdio.h>
#include "BigInt.h"

int main() {
        printf("Enter the 2 number: ");
        struct BigInt *val1 = _input();
        struct BigInt *val2 = _input();

        printf("their addition is ");
        _println(add(val1, val2));

        printf("their subtraction is ");
        _println(subtract(val1, val2));

        printf("their multiplication is ");
        _println(multiply(val1, val2));

        printf("their division is ");
        _println(divide(val1, val2));
        printf("35 raise to power 21 is ");
        _println(power(itob(35), itob(21)));

}
vagrant@myvm18:~$
```

OUTPUT

vagrant@myvm18: ~

```
vagrant@myvm18:~$ ./main.exe
Enter the 2 number: 212 23
their addition is 235
their subtraction is 189
their multiplication is 4876
their division is 9
35 raise to power 21 is 26633542255558204984664916992187 5
vagrant@myvm18:~$
```