Command line arguments

The following example shows how a program accepts and prints command line arguments. Program file name is considered as the first argument.

```
// C program named mainreturn.c to demonstrate the working
// of command line argument
#include <stdio.h>

// defining main with arguments
void main(int argc, char* argv[])
{
    if (argc==1){
        printf("\n No arguments given except file name");
        return;
    }
    printf("You have entered %d arguments:\n", argc);

    for (int i = 0; i < argc; i++) {
        printf("arg %d is %s\n", i, argv[i]);
    }
    return;
}
```

1. WAP to take 6 integers from the user as command line argument and short them in decreasing order.
2. WAP to take 5 names as command line arguments and print them into lexicographical increasing order. (Use <string.h>)

```
// C program for the above approach

#include <stdarg.h>
#include <stdio.h>

// Variadic function to add numbers
int AddNumbers(int n, ...)
{
        int Sum = 0;

        // Declaring pointer to the
        // argument list
        va_list ptr;
```

```
	// Initializing argument to the
	// list pointer
	va_start(ptr, n);

	for (int i = 0; i < n; i++)

		// Accessing current variable
		// and pointing to next one
		Sum += va_arg(ptr, int);

	// Ending argument list traversal
	va_end(ptr);

	return Sum;
}

// Driver Code
int main()
{
	printf("\n\n Variadic functions: \n");

	// Variable number of arguments
	printf("\n 1 + 2 = %d ",
		AddNumbers(2, 1, 2));

	printf("\n 3 + 4 + 5 = %d ",
		AddNumbers(3, 3, 4, 5));

	printf("\n 6 + 7 + 8 + 9 = %d ",
		AddNumbers(4, 6, 7, 8, 9));

	printf("\n");

	return 0;
}
```