String to Integer (atoi)

Program:

```
#include <stdio.h>
#include <ctype.h>
#include <limits.h>

int myAtoi(char *s) {
    int i = 0, sign = 1;
    long result = 0;

    // Skip leading whitespace
    while (isspace(s[i])) {
        i++;
    }

    // Check for the sign
    if (s[i] == '-' || s[i] == '+') {
        sign = (s[i] == '-') ? -1 : 1;
        i++;
    }

    // Convert digits to integer
    while (isdigit(s[i])) {
        result = result * 10 + (s[i] - '0');

        // Handle overflow and underflow
        if (sign == 1 && result > INT_MAX) {
            return INT_MAX;
        }
```

```
        if (sign == -1 && -result < INT_MIN) {
            return INT_MIN;
        }

        i++;
    }

    return (int)(sign * result);
}

int main() {
    char s1[] = "42";
    char s2[] = "   -42";
    char s3[] = "4193 with words";
    char s4[] = "words and 987";
    char s5[] = "-91283472332";

    printf("Converted integer: %d\n", myAtoi(s1)); // 42
    printf("Converted integer: %d\n", myAtoi(s2)); // -42
    printf("Converted integer: %d\n", myAtoi(s3)); // 4193
    printf("Converted integer: %d\n", myAtoi(s4)); // 0
    printf("Converted integer: %d\n", myAtoi(s5)); // -2147483648 (INT_MIN)

    return 0;
}
```