String to Integer (atoi)

Problem

Implement atoi which converts a string to an integer.

The function first discards as many whitespace characters as necessary until the first non-whitespace character is found. Then, starting from this character, takes an optional initial plus or minus sign followed by as many numerical digits as possible, and interprets them as a numerical value.

The string can contain additional characters after those that form the integral number, which are ignored and have no effect on the behavior of this function.

If the first sequence of non-whitespace characters in str is not a valid integral number, or if no such sequence exists because either str is empty or it contains only whitespace characters, no conversion is performed.

If no valid conversion could be performed, a zero value is returned.

Note:
Only the space character ‘ ’ is considered as whitespace character.
Assume we are dealing with an environment which could only store integers within the 32-bit signed integer range: $[-2^{31}, 2^{31} - 1]$. If the numerical value is out of the range of representable values, INT_MAX ($2^{31} - 1$) or INT_MIN (-2^{31}) is returned.

example 1

Input: ‘42’
Outpur: 42

example 2

Input: ” -42”
Outpur: -42

example 3

Input: “4193 with words”
Outpur: 4193

example 4

Input: “words and 987”
Outpur: 0

example 5

Input: “-91283472332”
Outpur: -2147483648

Solution

Consider all the cases:

Java

```
class Solution {
    public int myAtoi(String str) {
        Boolean whitespace = true;
        Boolean sign = true;
        Boolean start = true;
        String numstr = "";
        for(int i=0;i<str.length();i++){
            if(whitespace && str.charAt(i) == ' ')
                continue;
            if(whitespace && str.charAt(i) != ' ')
                whitespace = false;
            if(whitespace==false){
                if(str.charAt(i)!='-' && str.charAt(i)!='+' && (str.charAt(i)<'0' ||
str.charAt(i)>'9'))
                    break;
                if((sign==false || start==false) && (str.charAt(i)=='-' ||
str.charAt(i)=='+'))
                    break;
                if(sign && start && (str.charAt(i)=='-' || str.charAt(i)=='+')){
                    sign = false;
                    numstr += str.charAt(i);
                }
                if(str.charAt(i)>='0' && str.charAt(i)<='9'){
                    start = false;
                    numstr += str.charAt(i);
                }
            }
        }
        Boolean negative = false;
        int num = 0;
        int res = 0;
        for(int i=0;i<numstr.length();i++){
            if(numstr.charAt(i)=='-'){
                negative = true;
                continue;
            }
            else if(numstr.charAt(i)=='+')
                continue;
            else{
                if(negative){
                    if(res>Integer.MAX_VALUE/10 || (res==Integer.MAX_VALUE/10 &&
numstr.charAt(i)> '8'))
                        return Integer.MIN_VALUE;
                    else{
                        num = ((int) numstr.charAt(i) - 48);
                        res *= 10;
                        res += num;
                    }
                }
                else{
                    if(res>Integer.MAX_VALUE/10 || (res==Integer.MAX_VALUE/10 &&
numstr.charAt(i)> '7'))
                        return Integer.MAX_VALUE;
                    else{
                        num = ((int) numstr.charAt(i) - 48);
                        res *= 10;
                        res += num;
                    }
                }
            }
        }
```

```
        if(negative)
            res = res * -1;
        return res;
    }
}
```

Python

```
class Solution(object):
    def myAtoi(self, str):
        """
        :type str: str
        :rtype: int
        """
        whitespace = True
        sign = True
        start = True
        numstr = ""
        for i in range(len(str)):
            if whitespace and str[i] == ' ':
                continue
            if whitespace and str[i] != ' ':
                whitespace = False
            if whitespace == False:
                if str[i] != '-' and str[i] != '+' and (str[i] < '0' or str[i] > '9'):
                    break
                if (not(sign and start)) and (str[i] == '-' or str[i] == '+'):
                    break
                if sign and start and (str[i] == '-' or str[i] == '+'):
                    sign = False
                    numstr += str[i]
                if str[i] >= '0' and str[i] <='9':
                    start = False
                    numstr += str[i]

        negative = False
        res = 0
        for i in range(len(numstr)):
            if numstr[i] == '-':
                negative = True
                continue
            elif numstr[i] == '+':
                continue
            else:
                if negative:
                    if res > 214748364 or (res == 214748364 and numstr[i] > '8'):
                        return -2147483648
                    else:
                        num = int(numstr[i])
                        res *= 10
                        res += num
                else:
                    if res > 214748364 or (res == 214748364 and numstr[i] > '7'):
                        return 2147483647
                    else:
                        num = int(numstr[i])
                        res *= 10
                        res += num
        if negative:
            res = res * -1

        return res
```