Mining The Web For Code Assistance

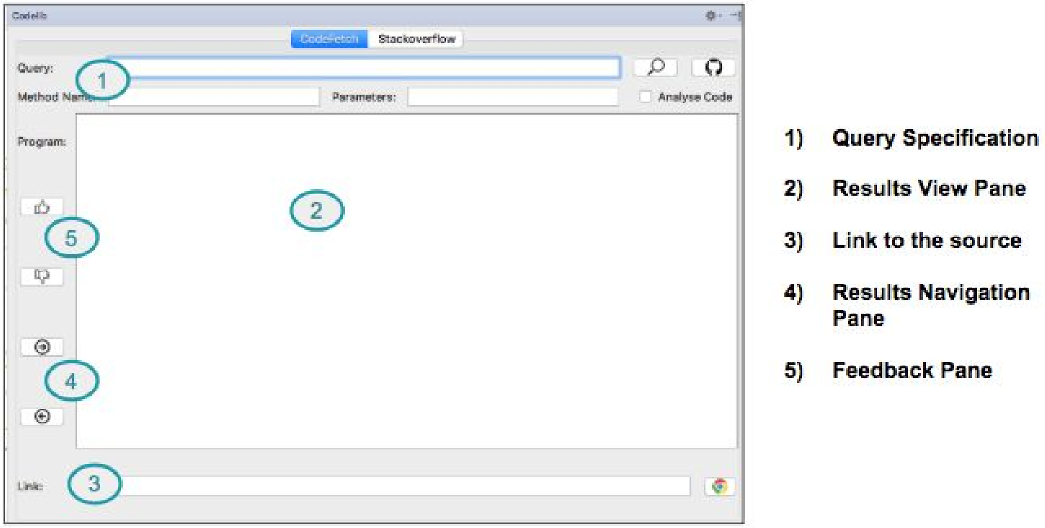
Aditya Parameswaran, Deva Surya Vivek Madala, Dibyendu Mishra
Mentors: Prof. Ashish Surekha, Anjali
CS-312, Program Analysis
Ashoka University

1.Introduction

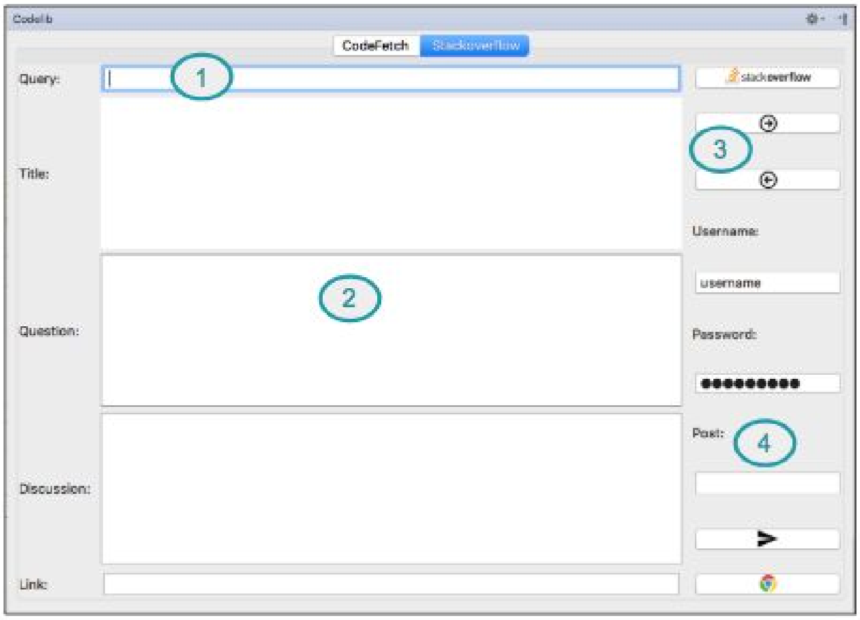
Although existing IDEs (e.g., Eclipse, Netbeans, VisualStudio) are equipped with various debugging supports for programming errors and exceptions, software developers often look into the web for working solutions and for any up-to-date information. According to the study of Brandt et al. ,developers spend about 19% of their development time in web surfing.

They resort to finding solutions to programming errors and exceptions or reviewing code on online QandA sites like StackOverflow, portals like GeeksForGeeks or development platforms like Github despite the availability of various debugging tools on their IDE. Nevertheless, these services are detached from the development environments used by programmers. Developers have to tap into this crowd knowledge through web browsers and cannot smoothly integrate it into their workflow. This situation hinders part of the benefits of these services.
We introduce CodeBuddy, a plugin for the IntelliJ IDEA IDE which provides developers access to these resources without having to leave their IDE. In this preliminary version our plugin is able to (1) retrieve source code from knowledge repositories like GeeksForGeeks for a given query (2) display relevant questions and discussions on StackOverflow for the query (3) start a new discussion on StackOverflow if no relevant discussion is found.

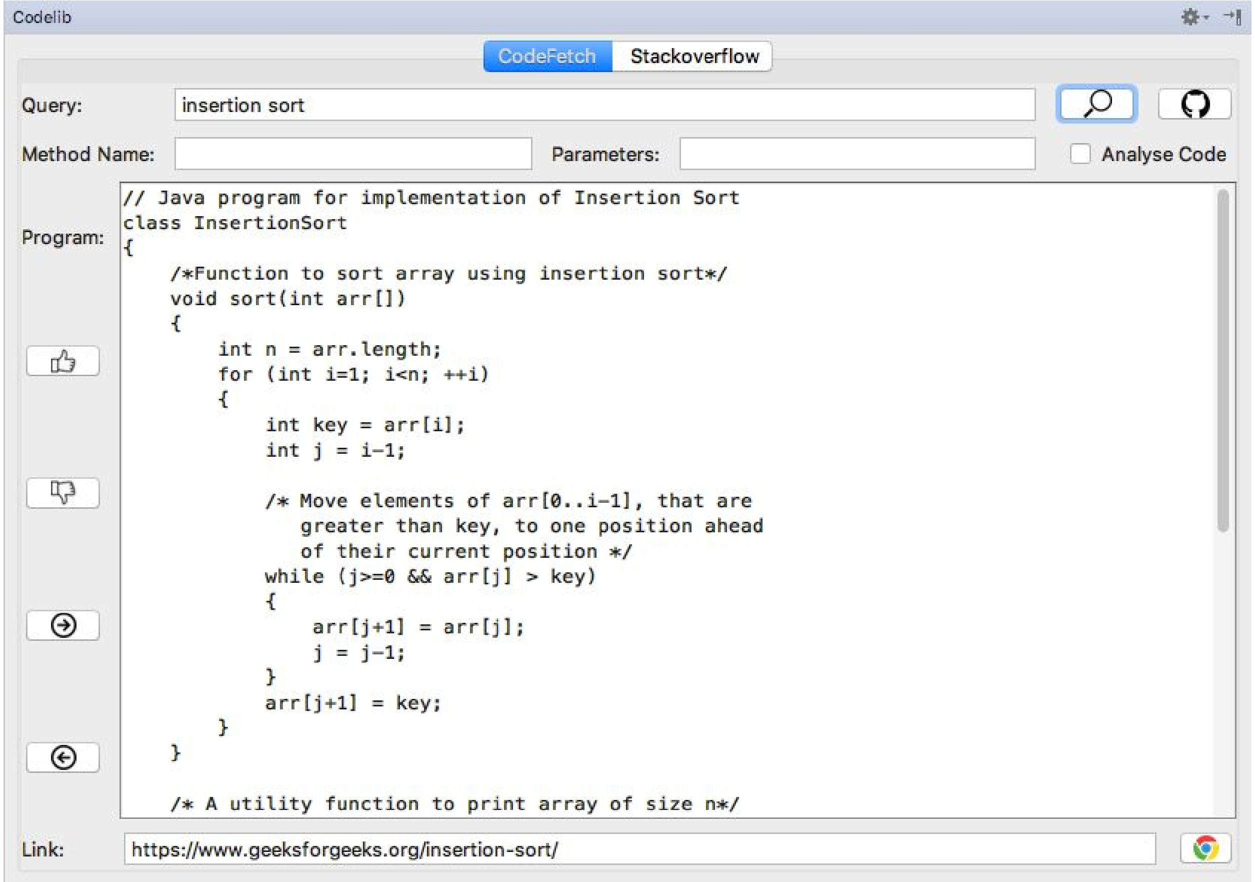
2.CodeBuddy



1) Query Specification
2) Results View Pane
3) Link to the source
4) Results Navigation Pane
5) Feedback Pane



1) **Query Specification Pane**
2) **Results Pane**
3) **Navigation Pane**
4) **Post a new question on StackOverflow**



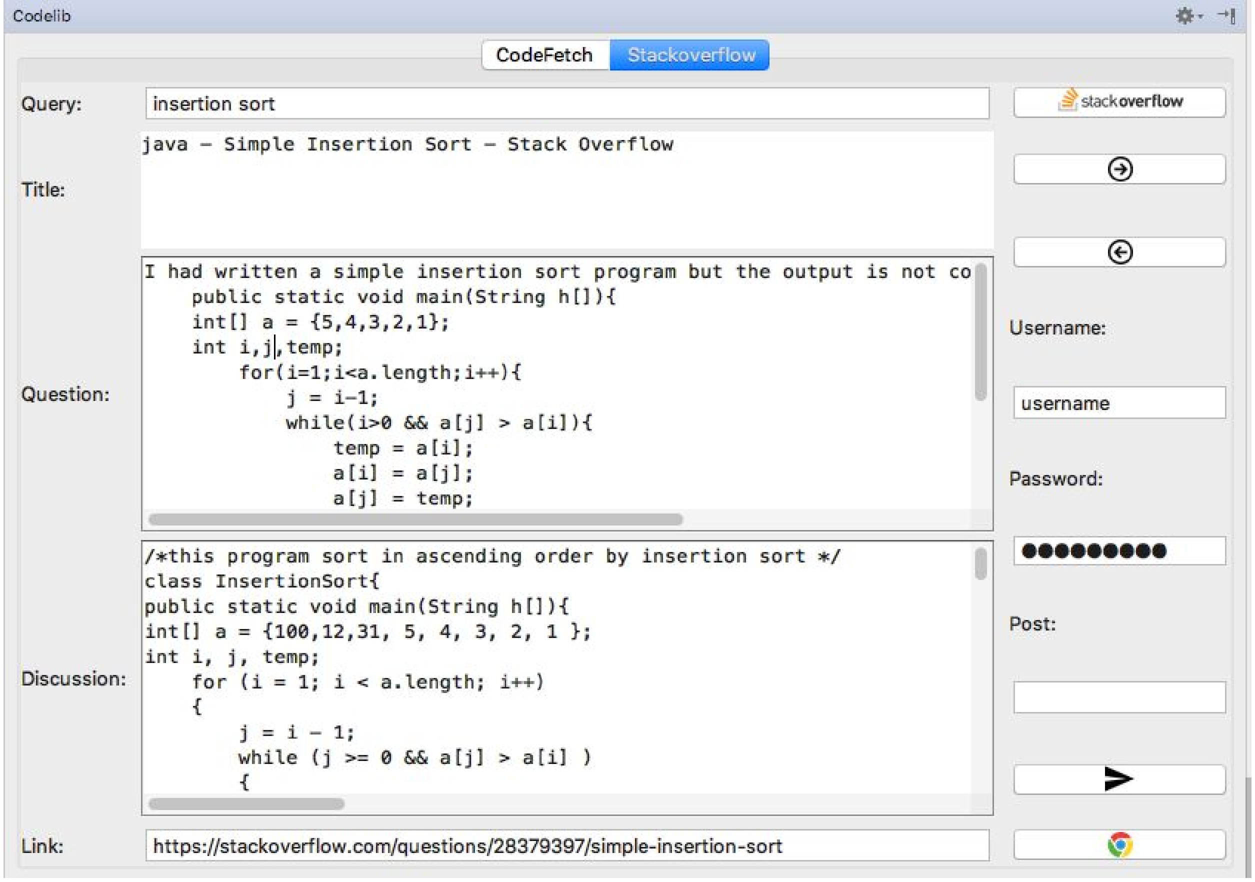
Query: insertion sort

```
// Java program for implementation of Insertion Sort
class InsertionSort
{
    /*Function to sort array using insertion sort*/
    void sort(int arr[])
    {
        int n = arr.length;
        for (int i=1; i<n; ++i)
        {
            int key = arr[i];
            int j = i-1;

            /* Move elements of arr[0..i-1], that are
               greater than key, to one position ahead
               of their current position */
            while (j>=0 && arr[j] > key)
            {
                arr[j+1] = arr[j];
                j = j-1;
            }
            arr[j+1] = key;
        }
    }

    /* A utility function to print array of size n*/
```

Link: https://www.geeksforgeeks.org/insertion-sort/



Query: insertion sort

Title: java - Simple Insertion Sort - Stack Overflow

Question:

```
I had written a simple insertion sort program but the output is not co
    public static void main(String h[]){
    int[] a = {5,4,3,2,1};
    int i,j,temp;
        for(i=1;i<a.length;i++){
            j = i-1;
            while(i>0 && a[j] > a[i]){
                temp = a[i];
                a[i] = a[j];
                a[j] = temp;
```

Discussion:

```
/*this program sort in ascending order by insertion sort */
class InsertionSort{
public static void main(String h[]){
int[] a = {100,12,31, 5, 4, 3, 2, 1 };
int i, j, temp;
    for (i = 1; i < a.length; i++)
    {
        j = i - 1;
        while (j >= 0 && a[j] > a[i] )
        {
```

Username: username

Password: ●●●●●●●●●●

Post:

Link: https://stackoverflow.com/questions/28379397/simple-insertion-sort

3. Performance

In order to evaluate our plugin we created a list of 46 queries with varying levels of complexity. We saw impressive results in terms of recall and precision using just the _ panel. The list of 46 queries are divided into 26 simple and 20 complex queries. The precision scores are calculated for each query and tabulated below. The overall precision is then deduced by averaging over the precision scores for all the queries. It is however noted that different features are used based on the type of queries. For instance, some complex queries require the use of additional features such as 'Search from Github' or 'Find Discussion on Stackoverflow' in order to get relevant results.

Sl. No.	Query	Precision	Type
1	Reversing a string	1	Easy
2	Reversing an array	1	Easy
3	Factorial of a number	0.875	Easy

4	Generate Fibonacci sequence	1	Easy
5	Sorting an array	1	Easy
6	Swapping two numbers	1	Easy
7	Find min and Max of an array	1	Easy
8	Add two numbers	0.66	Easy
9	Check for leap year	1	Easy
10	Check if a number is odd or even	1	Easy
11	Linear search	1	Easy
12	Generate random number	0.66	Easy
13	Sum of array	1	Easy
14	Average of numbers	1	Easy
15	Get input from user	1	Easy
16	Create a file	1	Easy
17	Read from a file	1	Easy
18	Write to a file	1	Easy
19	Create m x n matrix	1	Easy
20	Number of words in a sentence	1	Easy
21	Find ascii value of a character	1	Easy
22	Join two lists	1	Easy
23	Converting character array to string	1	Easy
24	Converting string to character array	0.75	Easy
25	Palindrome check	1	Easy
26	LCM of two numbers	1	Easy
27	Breadth first Search	1	Complex

28	Opening and closing a socket	1	Complex
29	Binary to decimal	1	Complex
30	Decimal to binary	1	Complex
31	Depth First Search	1	Complex
32	Dining philosophers	1	Complex
33	Find duplicate characters in a string	1	Complex
34	Term frequency in a sentence/paragraph	0.5	Complex
35	Sleeping Barber	0.5	Complex
36	String pattern matching	1	Complex
37	Kruskal’s Minimum Spanning Tree	1	Complex
38	Dijkstra's Shortest Path	1	Complex
39	Recursive Digit Sum	1	Complex
40	Caesar Cipher	1	Complex
41	Pangram Checking	0.66	Complex
42	Minimum Absolute Difference in an Array	1	Complex
43	Check if Sum and XOR are equal	1	Complex
44	Append to CSV	0.5	Complex
45	Check Prime Using Trial Division	0.75	Complex
46	Leonardo Numbers	1	Complex

Overall precision of the system = 42.855/46 = 0.931 (**93%** precision)

4.Future Development

- Using the feedback received by users in the form of upvotes and downvotes to improve solutions retrieved and provide better assistance to users across the platform.
- Form a codebase of our own by storing the results retrieved for each query such that machine learning algorithms can be implemented on the data to move towards automatic code generation as opposed to a purely retrieval based system.
- Creation of code templates so as to create structures that can invoked easily by the user for integrating with one's own program.
- Automatic query formation from active IDE context (by extracting keywords from the chosen code entities)

5.References

- Mining StackOverflow to Turn the IDE into a Self-Confident Programming Prompter https://www.si.usi.ch/assets/publications/conf/msr/msr2014/PonzanelliBPOL14.pdf

- SurfClipse: Context-Aware Meta Search in the IDE https://www.cs.usask.ca/~croy/papers/2014/MasudICSME2014ToolSurfClipse.pdf

- Discussions from Stackoverflow, special search on GitHub http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.700.5699&rep=rep1&type=pdf