Result Analysis on Survey: How do Developers Debug in Practice

This survey targets at understanding how developers debug in practice. Through this survey, we want to know what kind of information is useful for developers to debug in practice and the difficulties encountered by them in using such information. The results of this survey can direct the correct way for us to design automated debugging tools, which aims at facilitating developers in debugging automatically in practice.
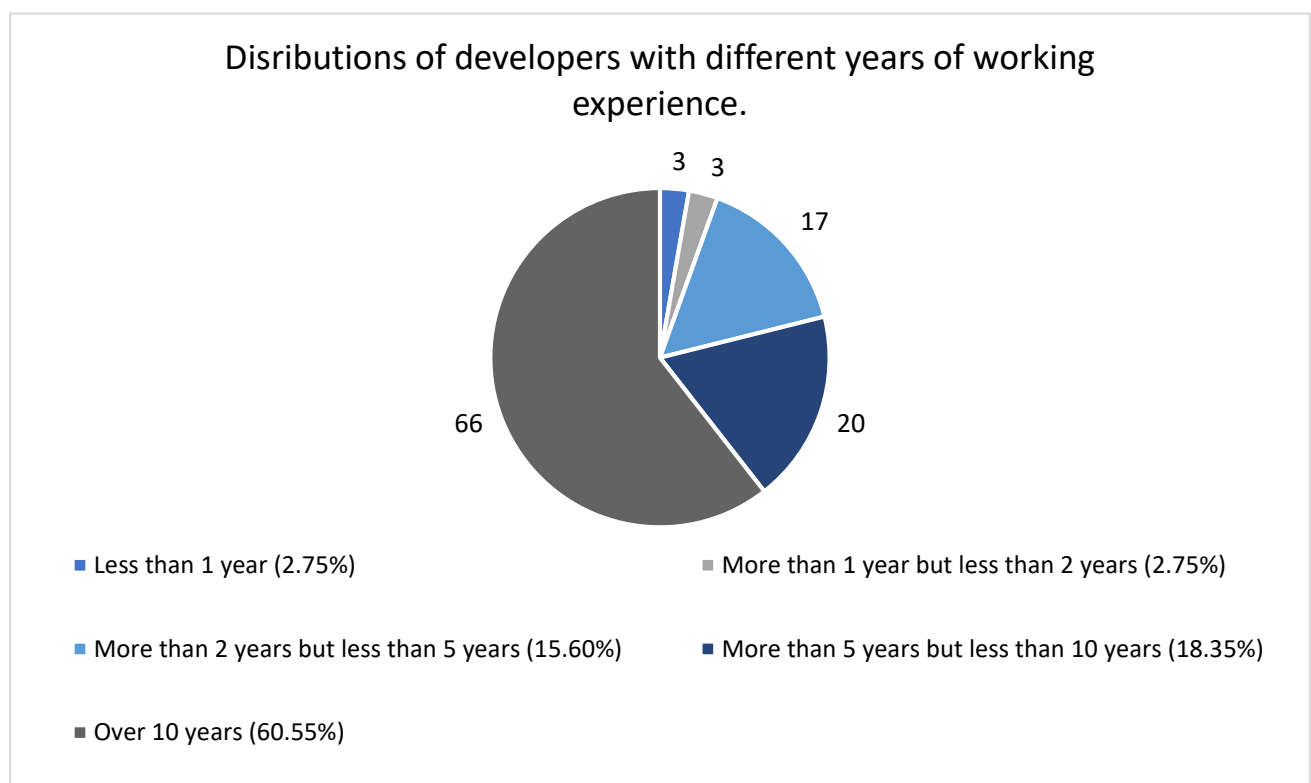
This survey contains only few questions, which can be finished within 2~4 minutes.
The link of this survey is: https://www.wjx.cn/jq/19791453.aspx

We conduct this survey in Microsoft. In total, we have received 109 valid feedbacks, and the following is the analysis of the results.
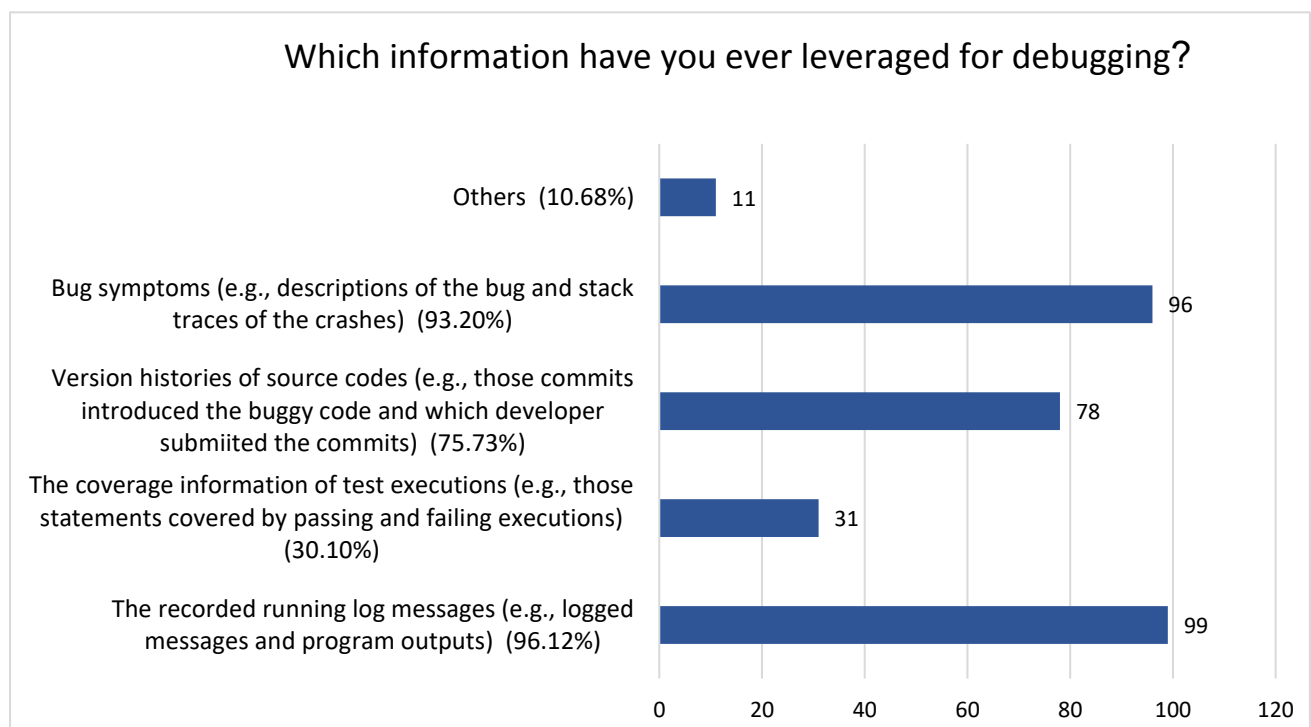
Distribution of developers with different years of developing experiences in industry.

We designed this question since the working experiences might vary for different developers, and this will affect the results of our survey. In this survey, around 60% of the developers have over 10-year working experience in industry and more than 78% interviewee have over 5-years experiences. We believe their opinions and feedbacks are representative among industries. To increase the confidence of our results, in the analysis of the following questions, we only those feedbacks submitted by developers with at least two years of developing experiences (103/109). We consider these developers are experienced in debugging

Disributions of developers with different years of working experience.

- Less than 1 year (2.75%)
- More than 1 year but less than 2 years (2.75%)
- More than 2 years but less than 5 years (15.60%)
- More than 5 years but less than 10 years (18.35%)
- Over 10 years (60.55%)

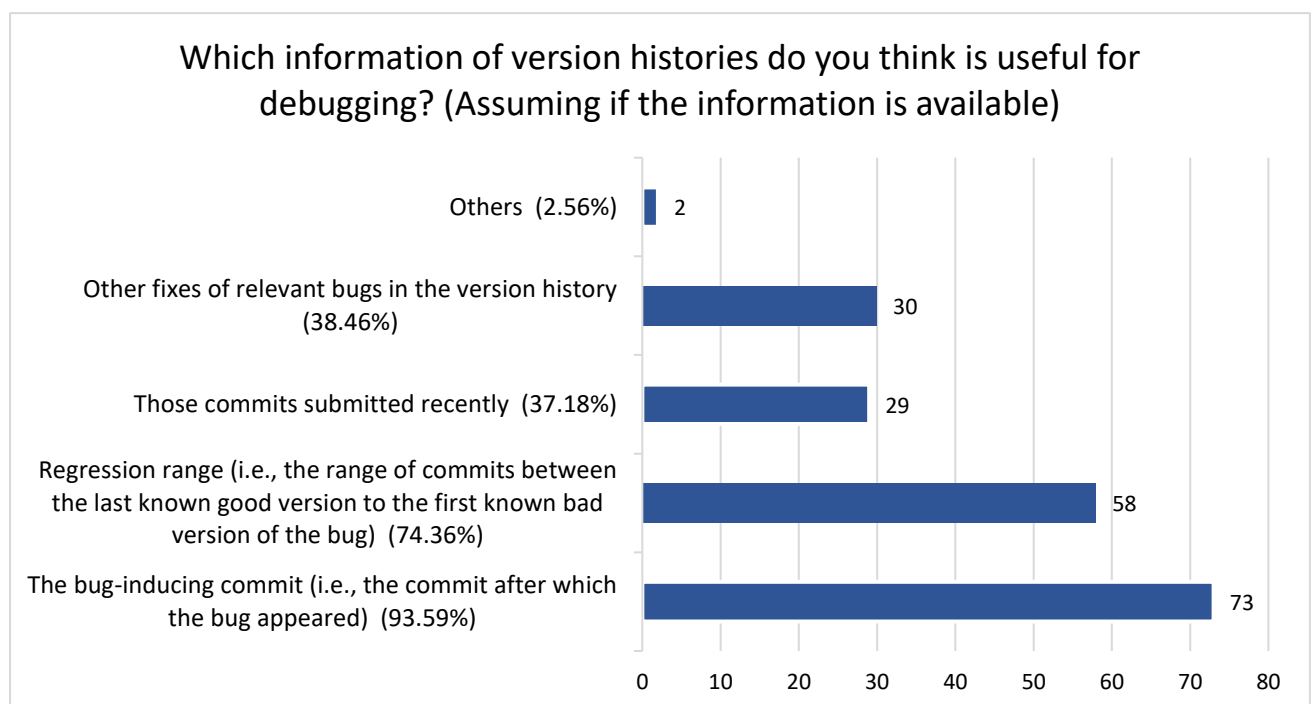## Q1. Which information have you ever leveraged for debugging?

All developers answered this multiple-choice question. In 103 feedbacks, around 95% developers have used **bug symptoms** and **recorded running log messages** for debugging. Besides that, three quarters of developers leveraged **version histories of source codes** for debugging. Surprisingly, the **test coverage** information of test executions is only leveraged by approximately 30% developers. From the results of this question, we can know that the information of version histories indeed is helpful for developer in debugging in practice

Which information have you ever leveraged for debugging?

Category	Value
Others (10.68%)	11
Bug symptoms (e.g., descriptions of the bug and stack traces of the crashes) (93.20%)	96
Version histories of source codes (e.g., those commits introduced the buggy code and which developer submiited the commits) (75.73%)	78
The coverage information of test executions (e.g., those statements covered by passing and failing executions) (30.10%)	31
The recorded running log messages (e.g., logged messages and program outputs) (96.12%)	99

Q2. Which information of version histories do you think is useful for debugging? (Assuming if the information is available).

This multiple-choice question is a follow-up question of Q1 and it was answered only by those developers who have chosen *version histories of source code* in Q1. In total, there are 78 developers answered this question.
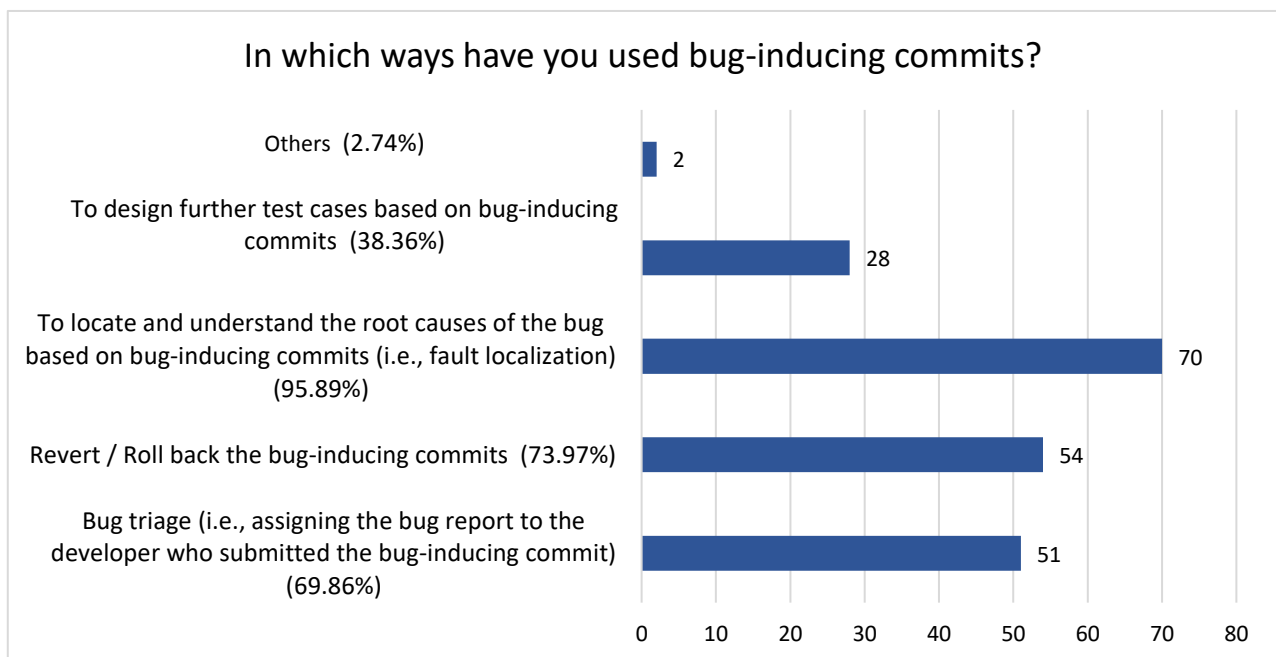
The bug-inducing commit obtained the highest vote (73/78) in the five options. *Regression range* is also considered as useful information of version histories by 74% (58/78) developers. Less than half developers thought the other two options are useful.

Which information of version histories do you think is useful for debugging? (Assuming if the information is available)

Option	Value
Others (2.56%)	2
Other fixes of relevant bugs in the version history (38.46%)	30
Those commits submitted recently (37.18%)	29
Regression range (i.e., the range of commits between the last known good version to the first known bad version of the bug) (74.36%)	58
The bug-inducing commit (i.e., the commit after which the bug appeared) (93.59%)	73

Q3. In which ways have you ever used bug-inducing commits?

This multiple-choice question is a follow-up question of Q2 and it will only be answered by those developers who have chosen **bug-inducing commits** in Q2. In total, there are 73 developers answered this question.

95.89% of the developers have ever leveraged the bug-inducing commits to **understand the root causes** of software bugs for **fault localization**. 73.97% of the developers have reverted the bug-inducing commits for debugging. 60% of them used it for **bug triage**. Moreover, 38.36% of the developers have ever used the bug-inducing commits in **designing extra test cases**. From the results, we can tell that almost all the developers have ever leveraged the bug-inducing commits in fault localization.

In which ways have you used bug-inducing commits?

| Category	Value
Others (2.74%)	2
To design further test cases based on bug-inducing commits (38.36%)	28
To locate and understand the root causes of the bug based on bug-inducing commits (i.e., fault localization) (95.89%)	70
Revert / Roll back the bug-inducing commits (73.97%)	54
Bug triage (i.e., assigning the bug report to the developer who submitted the bug-inducing commit) (69.86%)	51

Q4. Do you have tool support to leverage bug-inducing commits in fault localization? (except for those version control tools such as Git)

This multiple-choice question is a follow-up question of Q3 and it will only be answered by those developers who have chosen *fault localization* in Q3. In total, there are 70 developers answered this question.

Version control system, e.g., Git and SVN, are excluded in this question, since they are not designed specific for fault localization.

From the results, we can tell that a tool to support to leverage bug-inducing commits in fault localization is in highly demanded since 74% of the developers have to conduct the process of fault localization manually. Around 25% of them mentioned they have such tools such as VSTS.

Do you have tool support to leverage bug-inducing
commits in fault localization ?

18

52

■ No (74.29%)　　■ Yes (25.71%)
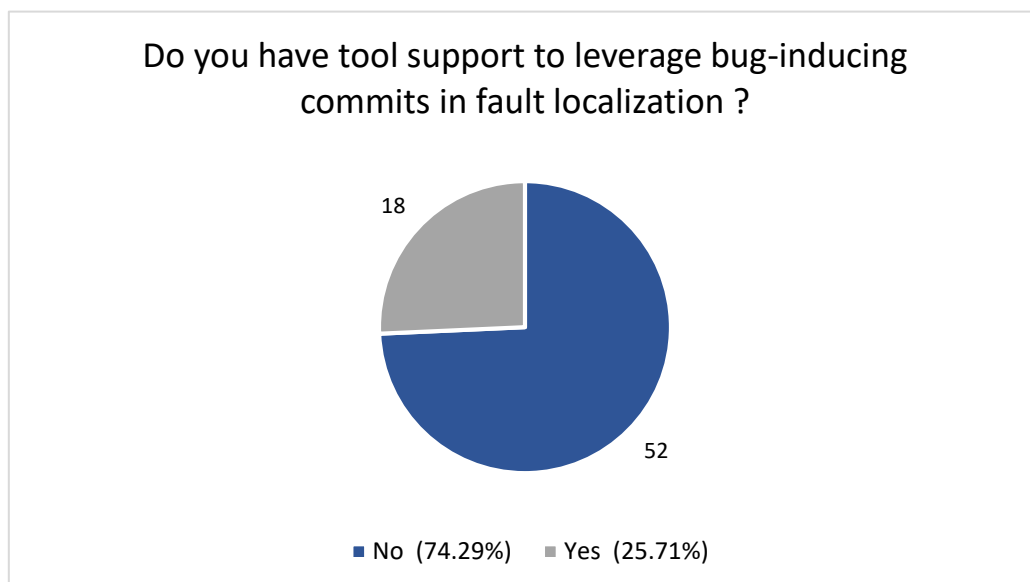
Q5. Why do you think bug-inducing commits is not useful for debugging?

This multiple-choice question is a follow-up question of Q2 and it will only be answered by those developers who have not chosen **bug-inducing commits** in Q2. In total, there are only 5 developers that answered this question. The results of this question may not be representative due to the small number of valid feedbacks.

Hard to obtain the bug-inducing commits and *lack of tool support* are the major concerns from developers.
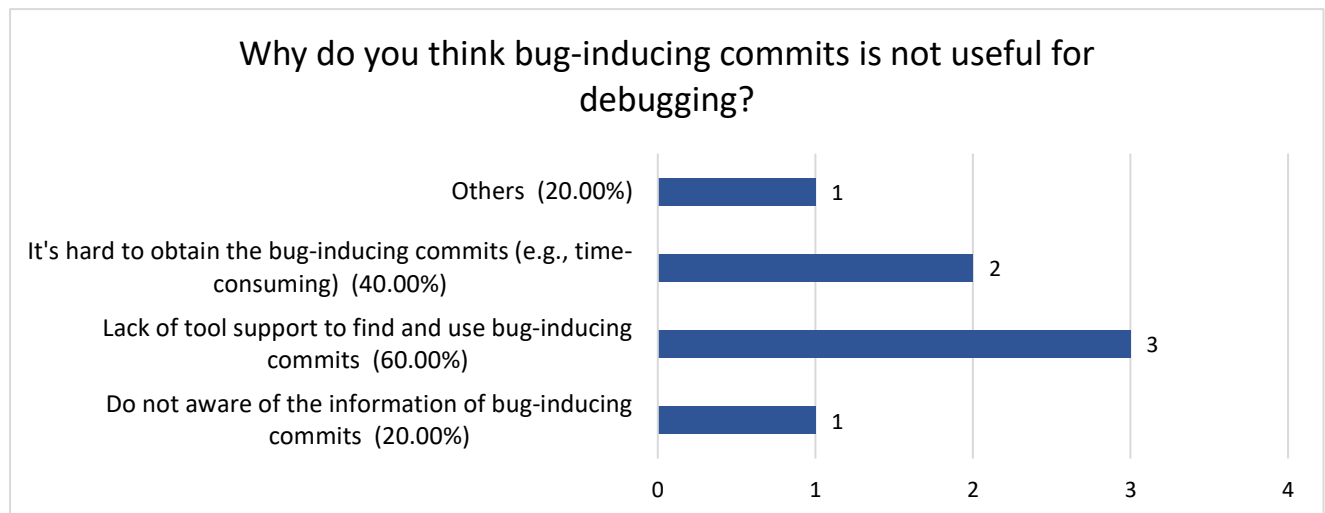
Q6. Why have not you ever leveraged version histories for debugging?

This multiple-choice question is a follow-up question of Q1 and it will only be answered by those developers who have not chosen **version histories of source code** in Q1. In total, there are 25 developers who answered this question.

Around 45% interviewees think the version history is not useful due to **lack of efficient methods or tool supports to leverage the information of version history**. And 18.9% of them **did not aware of such information**.

Why have not you ever leveraged version histories for debugging?

Category	Value
Other (20.8%)	11
Lack of methods to efficiently use version history since it's too large (45.3%)	24
Lack of tool supports to use version history for debugging (43.4%)	23
Do not aware of the information of version history (18.9%)	10