

Pairing Interview Guideline

This is a very hands-on interview where we let the candidate extend their submission for the code problem. The goal is to assess the candidate's ability to write software effectively. Here's how a 45 minute interview could be **structured**:

1. Chit chat and maybe a warm-up question (5 mins)
2. Let candidate walk through their code submission and follow up with questions (10 mins)
3. Pair with candidate to extend the code (see “Extensions” below) (25 mins)
4. Wrap up (5 mins)

During the interview, try to **assess** at least the following broad areas:

- Is the candidate able to explain her solution clearly?
- Is the candidate aware of strengths and weaknesses in her solution?
- Is the candidate responding to feedback and open to change her solution?
- Is the candidate able to understand a new requirement and translate it into code?
- What's the candidate's approach to implementing new functionality?
- etc.

Extensions

Here are some suggestions how the problem could be extended during the interview:

C Should clear the canvas.

U	Should undo the last command (“undo-undo” or “undo-history”)
---	--

G x y [c1..cn] Should do a radial version of a "bucket fill" using the colours specified. Remaining space in area is filled with last colour. For example, if we executed the following command as the last step of the sample program run above:

```
$ G 10 3 a b c d e
-----
|eeeeeeeeedcdeeeexxxx|
|xxxxxxedcbcddeex  x|
|      xdcbabcdexxxxx|
|      xedcbcddeeeeeee|
-----
```