Reverse Outlining

We usually think of an outline as something to organize your ideas before you put them on paper. However, you can also do a reverse outline after your draft is written to make your ideas and arguments more clear, organized and coherent.

Reverse outlining is a revision strategy that reexamines the components of a finished paper to see whether they succeeded in supporting the main claim you made in your thesis (just like mathematical proofs work backward to demonstrate the validity of a certain mathematical concept). A reverse outline also helps you visualize how the parts of your argument fit together.

Creating a Reverse Outline

Use the attached table to set up your reverse outline.

1. First, identify your thesis statement, which is most likely found in the introduction of your paper. This is written at the top of the reverse outline table because every paragraph should contribute to the argument that your thesis is presenting.
2. Next, number all of the body paragraphs in your draft.
3. In the Intended Purpose column, write down the job that each paragraph is supposed to be doing in your draft. Why did you include this section, and how should it be contributing to your argument?
4. In the Actual Main Idea(s) column, write down the function that each paragraph is actually serving in your draft. What is the main idea of the paragraph as it is written? (Even though every paragraph should have one main idea, write down all of the important points that this paragraph contains.)

Using Your Reverse Outline

When you’re finished completing the table, you can check whether the Intended Purpose and Actual Main Idea(s) columns from top to bottom. As you analyze the organization and flow of your draft, consider the following:

1. Are your ideas fully developed? Can you revise or strengthen some of your paragraphs so that the “Actual Main Idea” aligns with the “Intended Purpose” more effectively?
2. Do all of the points fit with your thesis? Are there unnecessary paragraphs that can be eliminated from your draft? Should you revise the thesis to better encapsulate the topics and evidence that you discussed in your paper?
3. Are your points presented in the most logical order? Do your paragraphs lead into each other, or are the pieces of your argument scattered throughout the paper? Could you rearrange the body paragraphs to group similar arguments, eliminate gaps in reasoning or allow for more natural transitions to arise as you develop your argument?
4. Do you have more than one idea in a single paragraph? Would it make sense to move some content to another paragraph, or did you develop new ideas that deserve a paragraph of their own? Did you find new and unexplored connections between ideas that you can further develop?
**Thesis Statement:** “Most of the accomplishments in Gandhi’s life are because of his honesty. Gandhi’s honesty can be seen in three scenarios; an incident about cheating at school, his marriage, and his nonviolent philosophy”

<table>
<thead>
<tr>
<th>Paragraph</th>
<th>Intended Purpose</th>
<th>Actual Main Idea(s)</th>
</tr>
</thead>
</table>
| 1         | In this section I wanted to show how Gandhi’s refusal to cheat in school demonstrates a commitment to honesty even during his childhood. | • Gandhi didn’t listen to his professor who wanted him to cheat to get good grades.  
• In Indian culture, it’s very important to respect your elders.  
• By relying upon his own abilities, he showed respect for the work of others  
• Gandhi was able to think for himself, thus allowing him to achieve greatness.  
• Gandhi also didn’t listen to traditional sexist readings that he encountered in school.  
• He eventually realized that women should be independent. |
| 2         |                   |                     |
| 3         |                   |                     |
| 4         |                   |                     |

Separating the **Intended Purpose** and the **Actual Main Idea(s)** can be very challenging to do on your own. Schedule an appointment with the GCC for assistance: www.cmu.edu/gcc