**Module 6: Critical Thinking**

**Stepwise Refinement Approach**

Ryan Thompson

Colorado State University - Global

CSC 505

Dr. Gonzalez

21 July 2024

**Synopsis**

A Level Computer Science. (2023, June 4) states the stepwise refinement approach is defined as breaking down a problem into a series of steps and substeps . Sometimes this is referred to as a top down approach or design. In this module I made use of the stepwise approach in order to convert a numerical check amount into a text representation. This is done by breaking up the number into smaller pieces. My approach takes the first digit of the number(the base value) and leaves the remainder. At which point the first digit is converted to text and the process is restarted with the remainder. The process continues to loop on itself until there is no remainder. For example, the number 142 will first be broken up into 1 and 42. The numerical 1 turns into a text one. Then the process starts over with the remainder(42), which is split into 4 and 2. Finally, converted into the text “forty two” and added to our “one hundred”. Thus turning into the text representation “One hundred and forty two dollars”. Overall, the stepwise approach is a powerful way to recursively break down a problem into smaller pieces.

**Analysis**

When tackling problems or managing projects, choosing the right approach can significantly impact outcomes. Two common methodologies are the step-wise approach and other alternative approaches, such as iterative development. Understanding their differences can help in selecting the most appropriate strategy for a given situation.

Step-Wise Approach

The step-wise approach, often associated with structured programming and software development, involves breaking down a problem or project into a series of sequential steps or phases. This method emphasizes systematic progression through a series of clearly defined stages.

Advantages:

1. Clarity and Structure: Each step has a specific objective and is completed before moving on to the next, providing a clear path forward. This can enhance understanding and manageability, especially for complex problems.
2. Focused Execution: By tackling one step at a time, the approach allows for focused efforts on each phase, potentially leading to higher quality in each part of the project.
3. Ease of Tracking Progress: Progress can be easily monitored as each step is completed, making it simpler to identify where issues might arise.

Disadvantages:

1. Inflexibility: Once a step is completed, making changes can be challenging. This rigidity can be problematic in dynamic environments where requirements frequently change.
2. Delayed Feedback: Because the step-wise approach often involves completing one step before moving on, it can delay the feedback cycle, potentially leading to late discovery of issues.

### Iterative Development

Iterative development involves repeating cycles of development, where each iteration builds upon the previous one. This approach is often used in agile methodologies.

Advantages:

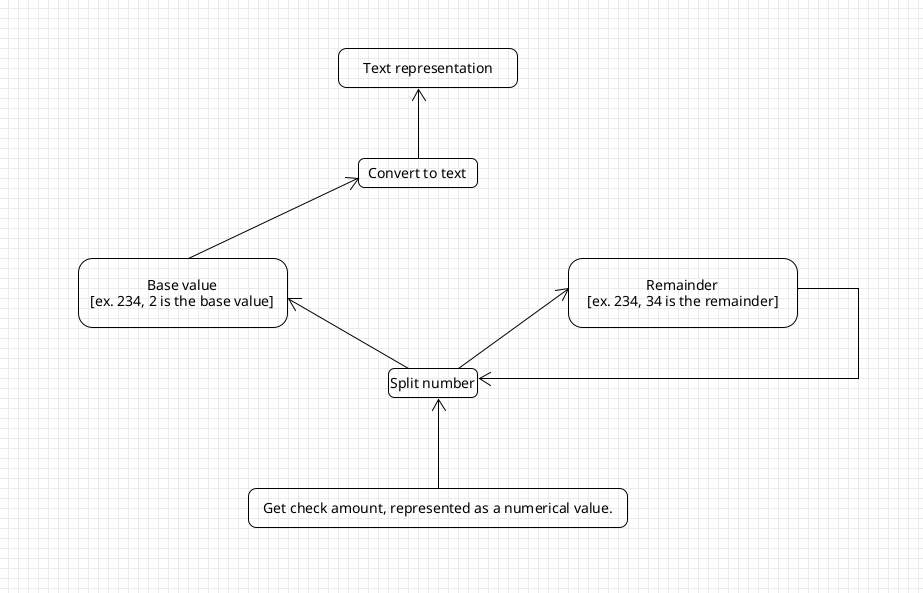
1. Flexibility and Adaptability: Iterations allow for ongoing adjustments based on feedback and changing requirements. This makes it easier to adapt to new information or shifting project needs.
2. Early Feedback: Each iteration provides an opportunity to test and get feedback on the developed features, which can lead to early detection of issues and alignment with user needs.
3. Incremental Progress: By delivering functional parts of the project in each iteration, stakeholders can see progress and provide input more frequently.

Disadvantages:

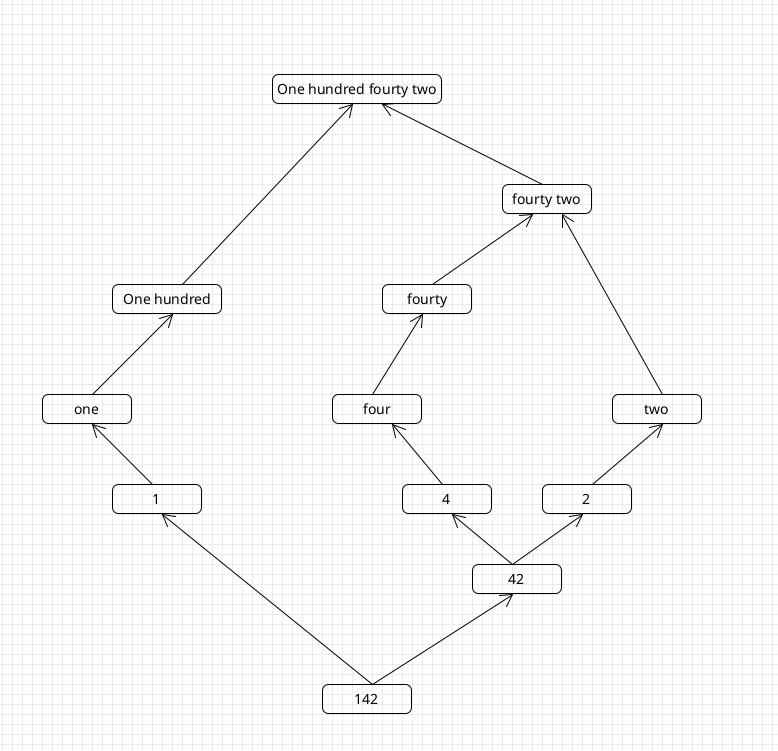
1. Potential for Scope Creep: Continuous changes and additions can lead to scope creep if not managed carefully, potentially causing delays or resource overruns.
2. Requires Frequent Reviews: Regular feedback and adjustments can be time-consuming and require ongoing involvement from stakeholders.

In conclusion, the choice between a step-wise approach and iterative development depends on the nature of the project, the level of required flexibility, and the need for early feedback. Each approach has its strengths and weaknesses, and understanding these can guide more effective project management and problem-solving strategies.

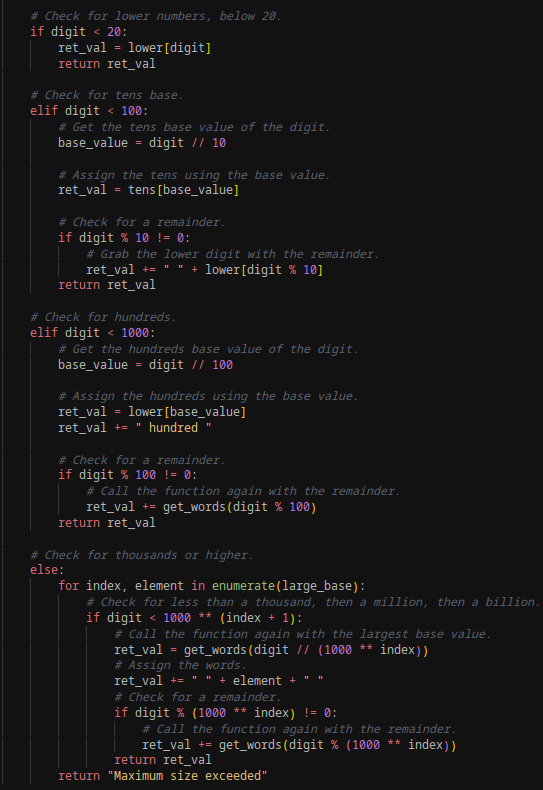
**Figures**



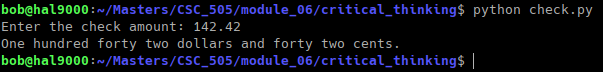
*UML Diagram*

**

*Example of 142 being converted into its text representation*

**

*Approach for splitting up the original numerical value*



*Program Output*

**References**

Stepwise refinement. A Level Computer Science. (2023, June 4). https://learnlearn.uk/alevelcs/stepwise-refinement/#google\_vignette

Stepwise refinement. (n.d.). https://www.cs.odu.edu/~zeil/cs333/website-f11/Lectures/stepwise/page/stepwise.html