

CPSC 410 Assignment 2

Exercise 1

- How could procedures(with or without parameters) be used in this language?
 - Our project aims to implement a DSL language called BSS which allows users to use variables or inheritance etc. when they are styling their websites rather than plain CSS which does not support these functionalities. The procedures in our projects could be used when the users need to do certain styling in multiple places, but don't want to write it multiple times. Then the users could define a procedure contains the way of styling and use it when they need.
- Do you think they would be beneficial for your users? Why, or why not?
 - This would benefit our users a lot. They could define many procedures about how they want to style the websites' elements in a certain way, and use these procedures later.

Exercise 3

1. True for both our language and ideal language. It follows orthogonality as we discussed in the class. We should be allowed to call procedures globally and locally (inside a procedure)
2. False for both our language and ideal language. If such procedures are allowed, programmers may call the procedures they would like to use later, so it is undefined procedures and a bad programming practice.
3. True for our language but false ideal language. For our language, Everytime we overwrite a new procedure we define before, we would also change the symbol table. Therefore, each time we call this procedure, we will call the "latest version" of it. However, for the ideal language, the programmers need to think what the procedures actually do since they may have same name but do different things.
4. It would be false for both our language and ideal language. It is the same reason as we talked in 3. If such behaviour is allowed, the programmers need to think about what the procedures actually do.
5. True for our language but false for ideal language. As it does not exist in most programming languages right now. It may cause confusion to programmers.

Exercise 7

- Briefly describe (in 2-3 sentences) at least one potential disadvantage of this feature (besides the additional implementation effort).
 - When the programming language supports the call-by-reference, the function could have potential adverse effects. The reason is that those functions may change the variables outside the scope of the the calling procedures, which may cause unexpected behaviours.
- Briefly describe why it is not possible to support call-by-reference without also introducing variable aliasing (even if there is no explicit alias keyword).

- As the code example given, the `b` in the procedure context and the `b` in the global context point to same location in the memory. They are aliases although we did not state "alias" in the code.