Report

Name: Jin Zhenming (김진명)

Student ID: 2022315690

This is the report for my PA3.

1. interp_expr

   I used pattern matching to process different types of AST nodes differently.

   - For number nodes (Num), it'll return the value of it.
   - For Add and Sub nodes (Add, Sub), it will recursively process two sides of the expression. Then return the result of the + or – operation to the results from two sides.
   - For Id nodes (Id), it'll search the name of the identifier in the storage (s) and return the corresponding value it has in the memory. If the name is unfound, the fails with an error message.
   - For LetIn nodes (LetIn), it will add a new identifier in the memory for the name with the evaluated value of the first expression. And evaluate the second expression with the updated memory.
   - For Call nodes (Call), first, it will check if the function being called exists (is already defined) in the function memory. If not, then fail with an error message. If

the function exists, then check if the number of parameters in the definition memory and the number of arguments in the function call matches (with List.length). If not matched, then it fails with a proper error message (organized with Printf.sprintf). Or the function call will be operated as normal, the function content will be operated.

- About normal function call:
  - arg_val is used to evaluate and store the value of each argument in the function call.
  - temp_para is used to store the parameters and their value for the function call. list.fold_left2 is used to access the list of arguments and bind them with parameters.
  - Then evaluate the function content with interp_expr.

2. interp_fundef
   - The only thing it does is store the function definition (function name, parameters, function content) in the function memory (FStore).

3. interp
   - This will first process the function definition (with interp_funde, “List.fold_left” is used to deal with function definitions which can access all definitions), and then it processes the expressions after the function definition (main_expr, with interp_expr).