Name: CS324 Key

CSE / IT 324 Assignment #1

Exercise 2-15:

Consider the following FORTRAN `SUBROUTINE`:

```
Line#
1     SUBROUTINE TEST (X, Y, Z)
2     X = 1
3     Z = X + Y
4     RETURN
5     END
```

and consider the following code fragment:

```
6     N = 2
7     CALL TEST (N, N, M)
```

What will be the final value of `M` if the parameters are passed by reference?

X and Y both point to N at the callee's Activation Record (AR). Thus, any change to either value (X or Y) will result in an immediate change to the value of N. The statement on line 2, 'X = 1', changes the value of N at the caller's side to be 1. Further, since both X and Y refer to this value of N,the statement on line 3, 'Z = X + Y', will result in M being assigned the value of '1 + 1' or 2.

What will be the final value of `M` if the parameters are passed by value-result?

When both parameters are passed by value, a formal copy is created, thus the statement on line 2, 'X = 1', will only alter the copy location of X, not the value of N. Thus, when the statement on line 3, 'Z = X + Y', is performed, we are adding the values of X and Y, where X = 1 and Y = 2. Hence, 3 will be assigned the value of '1 + 2' or 3.

Name: CS324 Key

Exercise 2-16:

There are several different varieties of pass by value-result: The address of the actual can be computed once, at subprogram entry time, or twice, once on entry and once on exit. Describe the output of this program under each of these two varieties of value-result:

```
Line#
1     DIMENSION A(2)
2     I = 1
3     A(1) = 10
4     A(2) = 11
5     CALL SUB (I, A(I))
6     PRINT, A(1), A(2)
7     END

8     SUBROUTINE SUB (K, X)
9     PRINT, X
10    K = 2
11    X = 20
12    RETURN
13    END
```

Description:

If the addresses are computed once, on entry at the caller, then the program will print out the values of 10, 20, and 11.When the statement on line 5 is performed, the value of the actual parameters, I and A(I), are 1 and 10, respectively. Inside of SUB, the formal-actual substitution process will place the value of 1 into the corresponding actual, K, and 10 into its corresponding actual, X. Thus, the statement on line 9 will print out the value 10. After executing lines 10 and 11, K and X will have the values of 2 and 20, respectively. When we return, the results are copied back into the actuals from the formals, thus assigning I the value 2 and A(1) the value, 20. The statement on line 6 will then print out 20 and 11.

If the addresses are computed on both entry and exit, then the program will print out the values of 10, 10, and 20. When the statement on line 5 is performed, the value of the actual parameters, I and A(I), are 1 and 10, respectively. Inside of SUB, the formal-actual substitution process will place the value of 1 into the corresponding actual, K, and 10 into its corresponding actual, X. Thus, the statement on line 9 will print out the value 10. After executing lines 10 and 11, K and X will have the values of 2 and 20, respectively. When we return, I will have the value 2, so when the addresses are recalculated, X will be assigned into A(2), not A(1). The statement on line 6 will then print out 10 and 20.

Does the outcome depend on the order in which the results are copied from the formals back into the actuals?

If the addresses are computed on entry and exit, yes. In the above example, if the actuals to formals were copied from right to left, I would still be 1 when X is being copied back into A(I), thus, X, which is 20, would be assigned into A(1). K would then be copied back into I. The statement on line 6 would then result in 20 and 11, which is a different result from above.