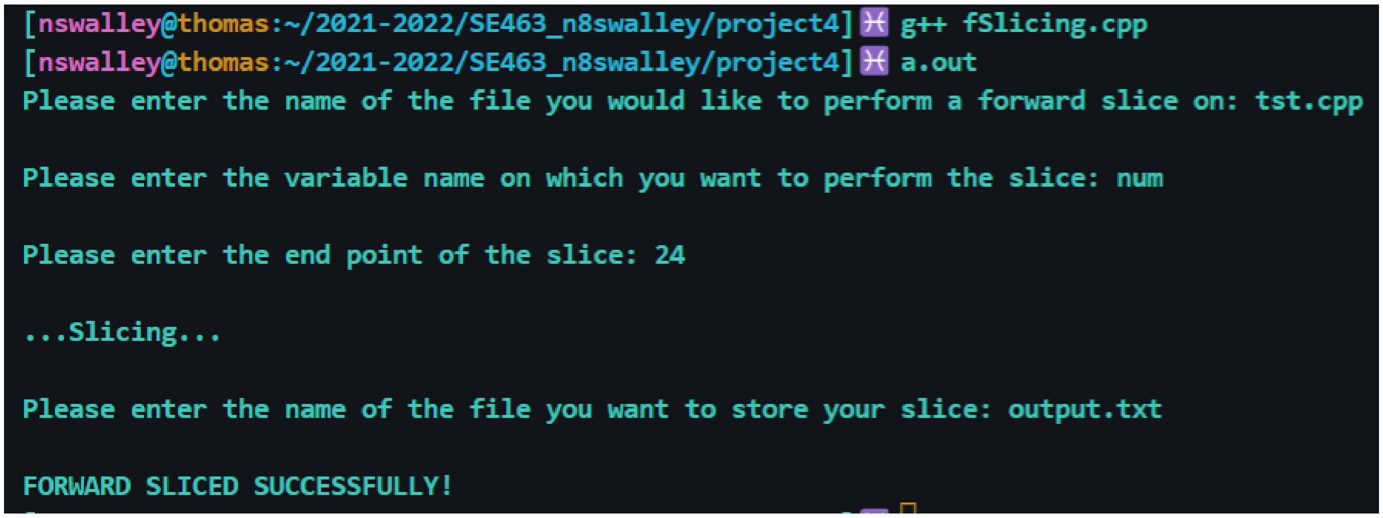
N8 SWALLEY 4/30/2022

SE463 – Spring 2022 Assignment #4 Forward Slicing

Introduction

This document outlines a C++ program called fSlicing.cpp that performs forward slicing on any given program. The program allows the user to provide a file name along with a variable name and an end point of the slice and then produces an output file with the specified forward slice. Forward slicing is a technique used in software testing which takes a slice or a group of program statements in the program for testing particular cases that may affect a value at a particular point of interest. It can also be used for the purpose of debugging in order to find issues more easily and efficiently.

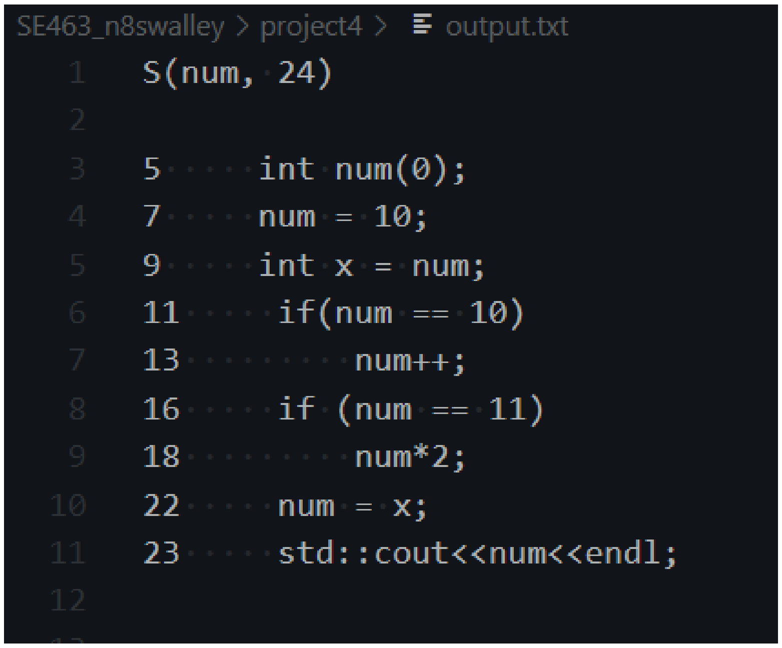
Sample Run:



```
[nswalley@thomas:~/2021-2022/SE463_n8swalley/project4]♓ g++ fSlicing.cpp
[nswalley@thomas:~/2021-2022/SE463_n8swalley/project4]♓ a.out
Please enter the name of the file you would like to perform a forward slice on: tst.cpp

Please enter the variable name on which you want to perform the slice: num

Please enter the end point of the slice: 24

...Slicing...

Please enter the name of the file you want to store your slice: output.txt

FORWARD SLICED SUCCESSFULLY!
```

Sample Program and Output:

In order to demonstrate that forward slicing works successfully in fSlicing.cpp, I created a sample program to test it on. The sample program was called tst.cpp and contained a variable named num that was defined and used throughout. Here is a screenshot of the output file produced by fSlicing.cpp after forward slicing on the sample program:

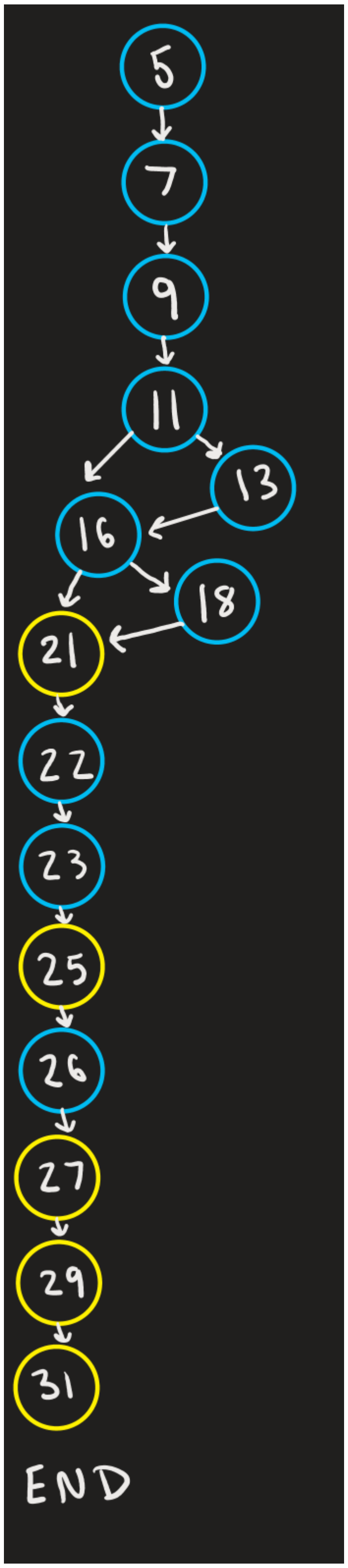


```
S(num, 24)

5    int num(0);
7    num = 10;
9    int x = num;
11     if(num == 10)
13         num++;
16     if (num == 11)
18         num*2;
22     num = x;
23     std::cout<<num<<endl;
```

Forward Slicing Benefits:

Forward Slicing is very beneficial in software testing. Since the lines at which variables are defined and used are the most likely to be the points that cause errors inside a program, forward slicing offers a quick and easy way to identify those key areas. Forward slicing is very beneficial for debugging because it simply points to the specific areas where bugs are possible. This program, fSlicing.cpp, can easily identify which lines contain the variable to inspect and allows for further analysis of the DEF and USE nodes of that specified variable.

Sample Program Graph:



Each blue node indicates lines that involve the num variable in the sample program. This program graph makes it much simpler to label each node as either a DEF or USE node. DEF nodes of a variable demonstrate when the variable's value is affected while USE nodes of a variable are when the variable's value is being used within the program. Once again, forward slicing is very beneficial because it displays all of the lines that these types of nodes would be on and makes it much easier to identify them within the program.

DEF and USE Nodes:

From the output of fSlicing.cpp, I was able to easily identify that the num variable (up to line 24), was used on several different lines. I was able to analyze the output and determine which lines are DEF and USE nodes. Here is a chart displaying those nodes:

Variable	**DEF Nodes**	**USE Nodes**
num	7, 13, 18, 22, 26	9, 11, 16, 23

I wrote the sample program in order to test the different types of USE and DEF nodes that could exist for the num variable. For the DEF nodes, I defined num with a hard-coded value (7), an incremental change (13), an operational change (18), the value of another variable (22) and a user input (26). For the USE nodes, I first used the value of num by assigning it to another variable (9). I also used its value within two different conditional statements (11,16) and printing it out (23). Each of these cases provide full coverage to my forward slicing program because they demonstrate the ways in which a variable can be defined and used. All in all, my forward slicing program is able to identify a program's nodes for a given variable and ending point, which ultimately allows the user to test each node as needed. The program allows users to take full advantage of the benefits forward slicing has to offer.