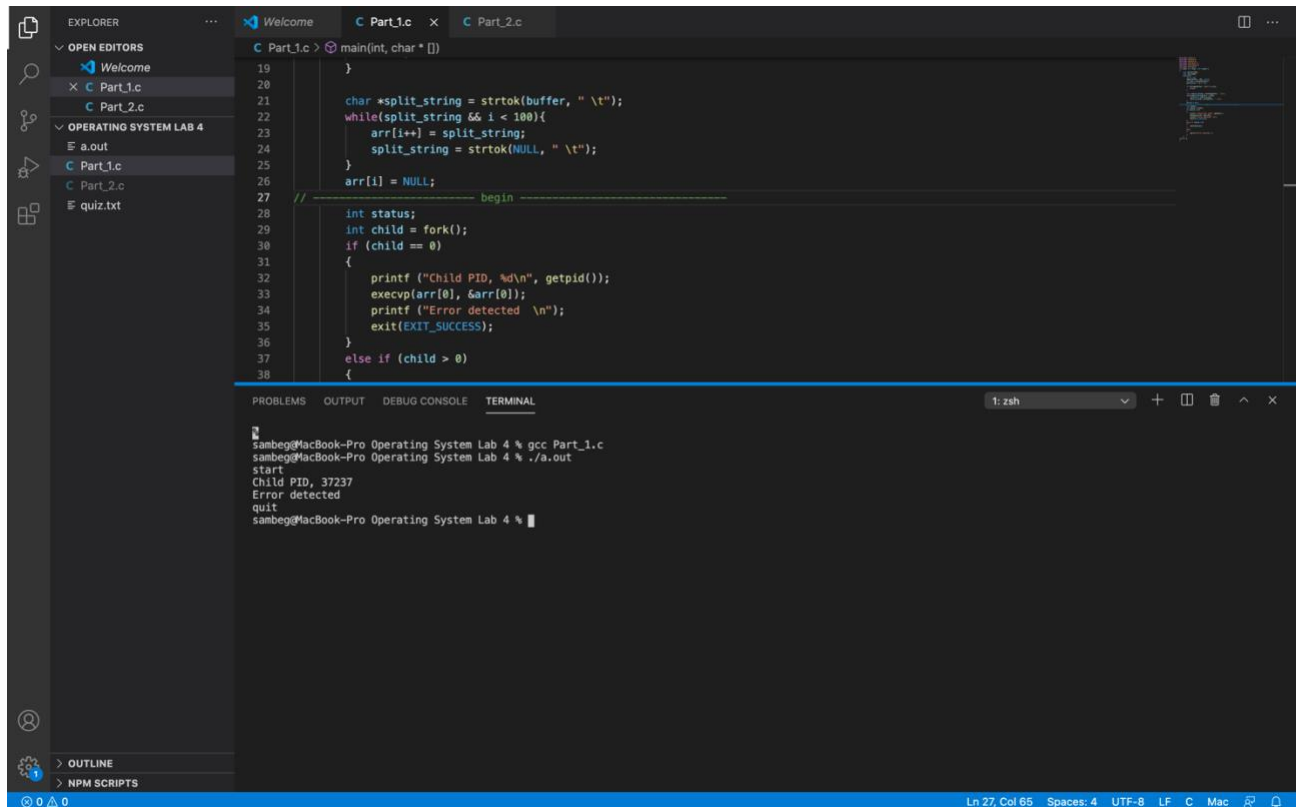


**Name: Sambeg Raj Subedi**

**Date: 04/08/2021**

**Lab 4 Report:**

**Part 1 Screenshot:**



The screenshot displays a Visual Studio Code editor window with a C program in `Part_1.c`. The program's logic is as follows: it defines a `main` function that takes `int` and `char*` arguments. It uses `strtok` to split the input buffer by spaces, storing the tokens in an array `arr`. A `while` loop processes these tokens. A `begin` comment marks the start of a `fork` operation. The parent process prints the child's PID and then calls `execvp` with the first token from the array. The child process prints "Error detected" and calls `exit(EXIT_SUCCESS)`. The parent process then checks if the child is still running.

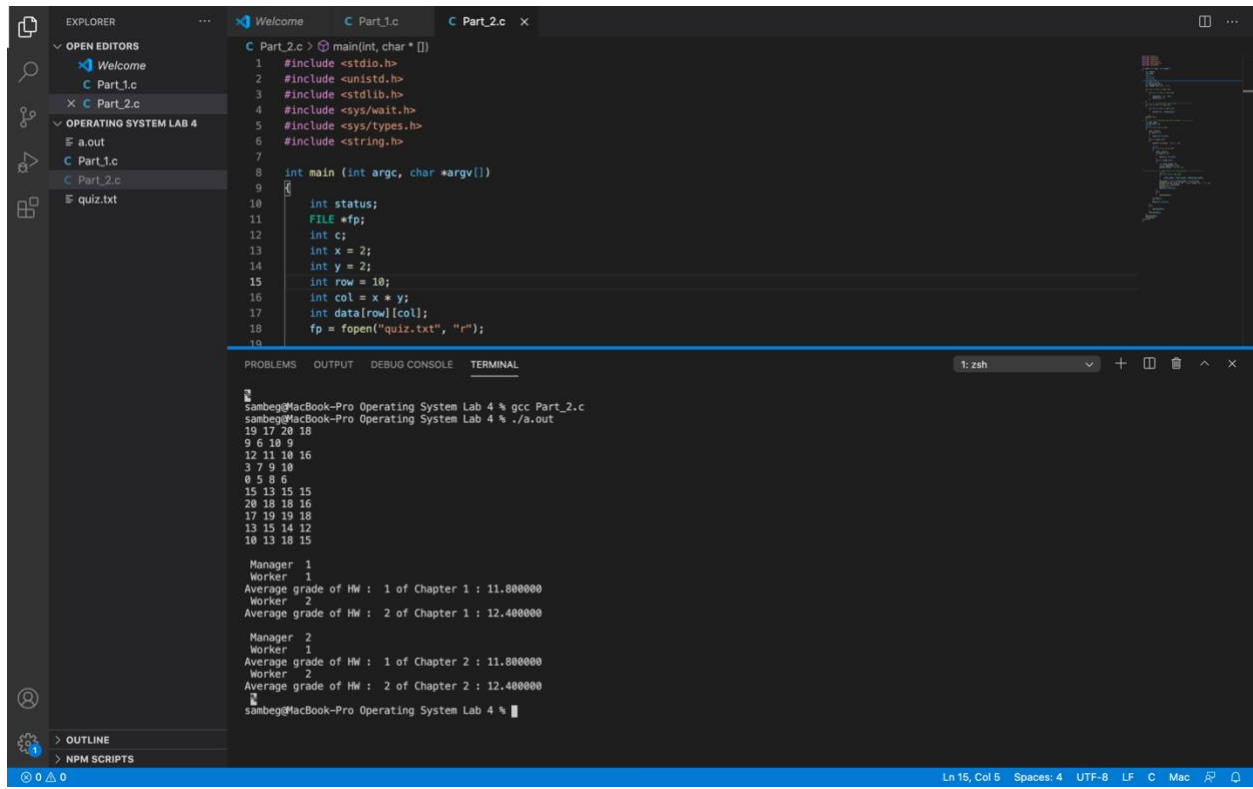
```
19 }
20
21 char *split_string = strtok(buffer, " \\t");
22 while(split_string && i < 100){
23     arr[i++] = split_string;
24     split_string = strtok(NULL, " \\t");
25 }
26 arr[i] = NULL;
27 // ----- begin -----
28
29 int status;
30 int child = fork();
31 if (child == 0)
32 {
33     printf("Child PID, %d\\n", getpid());
34     execvp(arr[0], &arr[0]);
35     printf("Error detected \\n");
36     exit(EXIT_SUCCESS);
37 }
38 else if (child > 0)
39 {
```

The terminal output at the bottom shows the execution of the program:

```
sambeg@MacBook-Pro Operating System Lab 4 % gcc Part_1.c
sambeg@MacBook-Pro Operating System Lab 4 % ./a.out
start
Child PID, 37237
Error detected
quit
sambeg@MacBook-Pro Operating System Lab 4 %
```

Fig: Output for Part 1.

## Part 2 Screenshot:



The screenshot shows a Visual Studio Code editor with a C program in `Part_2.c`. The program includes `<stdio.h>`, `<unistd.h>`, `<stdlib.h>`, `<sys/wait.h>`, `<sys/types.h>`, and `<string.h>`. The `main` function takes `argc` and `argv` as arguments. It declares `int status;`, `FILE *fp;`, `int c;`, `int x = 2;`, `int y = 2;`, `int row = 10;`, `int col = x * y;`, and `int data[row][col];`. It then opens `quiz.txt` for reading and prints its contents.

```
1 #include <stdio.h>
2 #include <unistd.h>
3 #include <stdlib.h>
4 #include <sys/wait.h>
5 #include <sys/types.h>
6 #include <string.h>
7
8 int main (int argc, char *argv[])
9 {
10     int status;
11     FILE *fp;
12     int c;
13     int x = 2;
14     int y = 2;
15     int row = 10;
16     int col = x * y;
17     int data[row][col];
18     fp = fopen("quiz.txt", "r");
```

The terminal output shows the execution of the program. It displays the contents of `quiz.txt` and the average grade of the homework for each chapter.

```
sambeg@MacBook-Pro Operating System Lab 4 % gcc Part_2.c
sambeg@MacBook-Pro Operating System Lab 4 % ./a.out
9 6 10 9
12 11 10 16
3 7 9 10
0 5 8 6
15 13 15 15
20 18 18 16
17 19 19 18
13 15 14 12
10 13 18 15

Manager 1
Worker 1
Average grade of HM : 1 of Chapter 1 : 11.800000
Worker 2
Average grade of HM : 2 of Chapter 1 : 12.400000

Manager 2
Worker 1
Average grade of HM : 1 of Chapter 2 : 11.800000
Worker 2
Average grade of HM : 2 of Chapter 2 : 12.400000
sambeg@MacBook-Pro Operating System Lab 4 %
```

Fig: Output for Part 2.