

## Assignment 3 Documentation

Q2

### 6 Register Sets

```
Microsoft Visual Studio Debug Console
Enter number of Register Sets: 6, 8, 16
6
Register Sets: 6
Computing Ackermann with x = 3 and y = 6
Procedural Calls: 172233
Max Depth: 511
Register Window Overflows: 84885
Register Window Underflows: 84885
Average Time Taken: 0.010405

C:\Users\ASUS\source\repos\Assignment3\Debug\Assignment3.exe (process 10220) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

### 8 Register Sets

```
Microsoft Visual Studio Debug Console
Enter number of Register Sets: 6, 8, 16
8
Register Sets: 8
Computing Ackermann with x = 3 and y = 6
Procedural Calls: 172233
Max Depth: 511
Register Window Overflows: 83911
Register Window Underflows: 83911
Average Time Taken: 0.010253

C:\Users\ASUS\source\repos\Assignment3\Debug\Assignment3.exe (process 2424) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

### 16 Register Sets

```
Microsoft Visual Studio Debug Console
Enter number of Register Sets: 6, 8, 16
16
Register Sets: 16
Computing Ackermann with x = 3 and y = 6
Procedural Calls: 172233
Max Depth: 511
Register Window Overflows: 80142
Register Window Underflows: 80142
Average Time Taken: 0.010285

C:\Users\ASUS\source\repos\Assignment3\Debug\Assignment3.exe (process 1292) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

Q2 As expected, that the procedure calls, maximum register window depth will be the same when the program is compiled and run with 3 different numbers of register sets. The frequency of overflows occurring when the program is runned also equals the frequency of underflows for all three cases. However, as the number of register sets increases, the frequency of overflows and underflows occurring decreases.

Q3 I have used time.h in the cpp library to measure the time taken to run ackermann(3, 6). This allows me to only calculate the time taken for the implementation more accurately by only begin the clock when the program runs and ending it after it finishes. Also, to further improve my accuracy, I ran the code 1000 times and get the average running time for it.