Compiler Construction
Problem Set 4
Alicja Jonczyk

1. **Translate the program into Three-Address Code (TAC)**

```
external func print(...)

func fibonacci(n, l, a, b, r, i)
    L:
        r = a + b
        a = b
        b = r
        i = i + 1
        if i < l goto L
    return r

func main()
    param 5
    param 6
    param 0
    param 1
    param 0
    param 1

    result = call fibonacci, 6

    param "The "
    param n
    param "th number in the sequence is: "
    param result

    call print, 4

    return 0

call main, 0

// print is a function that performes the concatenation of its parameters
```

2. **Symbol Table Creation**

```
Files suggested/locals.symbols and locals.symbols are identical
Files suggested/strings.symbols and strings.symbols are identical
Files suggested/shadowing.symbols and shadowing.symbols are identical
Files suggested/globals.symbols and globals.symbols are identical
No differences found in PS4!
jonczyk@alicja:~/COMPILER_CONSTRUCTION/problem_set_4/vsl_programs$
```

```
==8420== All heap blocks were freed -- no leaks are possible
==8420==
==8420== For lists of detected and suppressed errors, rerun with: -s
==8420== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
jonczyk@alicja:~/COMPILER_CONSTRUCTION/problem_set_4$
```