TDT4205 Compiler Construction
Assignment 4

1 Theory

1.1 Three Access Code and Stack Machine 10%

The following vsl code calculates the greatest common divisor (GCD/HCF) for the values and

```
def main() begin
var k = gcd(1144,546)
end

def gcd(a,b)
begin
var g
if b>0: then
g:=gcd(b,a-((a/b)*b))
else
g:=a
return g
end
```

1. Derive the TAC representation for the above code
2. Suggest a certain stack layout for the gcd function. on a machine which passes arguments through the runtime stack. Explain what each part in your stack does. You may use the X86 convention covered in the lecture.
3. Draw the stack layout when the gcd function is about to return to the main function.

2 Programming Exercise

The parts for both exercise 4 and 5 are released separately. This is the programming part for exercise 4.

2.1 15%

Implement the function find_globals in ir.c to populate the global symbol table with

1. Global Variables (5%)
2. Functions (10%)

2.2 45%

Implement the function bind_names in ir.c to populate the local symbol tables with

1. Parameters (15%)
2. Local Variables (15%)

and link the entry pointers in its syntax tree nodes to their appropriate symbols (15%)

2.3 10%

Extend bind_names to create a global table of strings.

2.4 5%

Implement the function destroy_symtab to remove dynamically allocated symbol table data at the end of the compilation.