

Seminar 7: Intermediate code generation

Exercise 1

Given the follow C code:

```
#include <stdio.h>

float pi;

float AreaCirculo (float pi, float r)
{
    float temp;
    temp = pi * r * r;
    return temp;
}

int main()
{
    float radio, sol;
    pi = 3.1416;
    printf("Introduzca el radio del circulo: ");
    scanf("%f", &radio);
    sol = AreaCirculo(pi, radio);
    printf("El area del circulo es %f\n, sol");
    return 0;
}
```

Show the assembler program the C code generates

Exercise 2

Intermediate code three addresses for **assignment statements**. Overall our instruction set will be as follows:

- Assignment statements of the form "X: = Y op Z", where op is a binary arithmetic operator (+, -, *, /)
- Assignment statements of the form "X: = op Y", where op is a unary operator
- Copy sentences: "X: = Y"

Design the translation scheme for three-address code for assignment statements corresponding to the syntax tree. A possible example is shown below:

Example:
 $A := B * -C + B * -C$

```
T1 := -C
T2 := B*T1
T3 := -C
T4 := B*T3
T5 := T2+T4
A := T5
```