

## CURSO DE COMPILADORES

### PROPUESTA DE LENGUAJE

#### LENGUAJE PARA GRAFOS

**Nombre:** Alexander Pinto De la Gala

Se desea implementar un lenguaje para resolver problemas simples de grafos, proporcionando un alto nivel de abstracción, de manera que el manejo de la estructura de datos de bajo nivel no esté por cuenta del usuario.

#### DESCRIPCIÓN DE ELEMENTOS Y EJEMPLOS DE USO

Se tendrá dos Clases principales de variables Node y Edge, las cuales a su vez contendrán un atributo (o peso en caso de Edge) de tipo int o string.

- Creación de variable  
node int A;                      creación de una variable de tipo nodo con atributo entero  
node int A = 10;                el atributo tomará el valor de 10;  
  
edge int z = 100                creación de una variable tipo Edge con peso igual a 100.
- Suma de atributos/peso  
int c = A + B                    suma de los atributos de los nodos A y B
- Comparación de atributos /peso  
>, <, ==, <=, >=, <>
- Asignación de Edge a Nodo  
z -> A
- Creación de Arrays de Nodos  
ARR = [A, B, ..., F]
- Iteración en Array  
for i in ARR  
{  
...  
}
- Acceso a edges de un nodo.  
iA
- Acceso a nodo de Edge (derecho/izquierdo)  
  
i < a    i > a
- Iteración de edges in Nodo  
for i in A  
{  
...  
}
- Condicional:  
if (i < x)  
{...}

```
else {...}
```

- Edge de menor valor en Nodo  
edge int x = 10000;  
for i in Nodo  
{  
  If (i < edge)  
  { edge = i;  
  }  
}  
- Elegir siguiente Nodo con edge de menor valor  
ARR = [A, B, C, D];  
Nodo int Nodo\_selec;  
for N in ARR:  
{  
  edge int x = 10000;  
  for i in N  
  {  
    If (i < edge)  
    { edge = i;  
      If (<edge == N)  
      Nodo\_selec = >edge;  
    else:  
      Nodo\_selec = <edge;  
    }  
  }  
}