Trabalho Final

Linguagem de Programação II e Estruturas de Dados II

Grupo:

- ➔ Deyvid William Silva de Medeiros
- ➔ Guilherme Euller Dantas Silva
- ➔ Nathãn Vitor de Lima

Diagrama de Classes

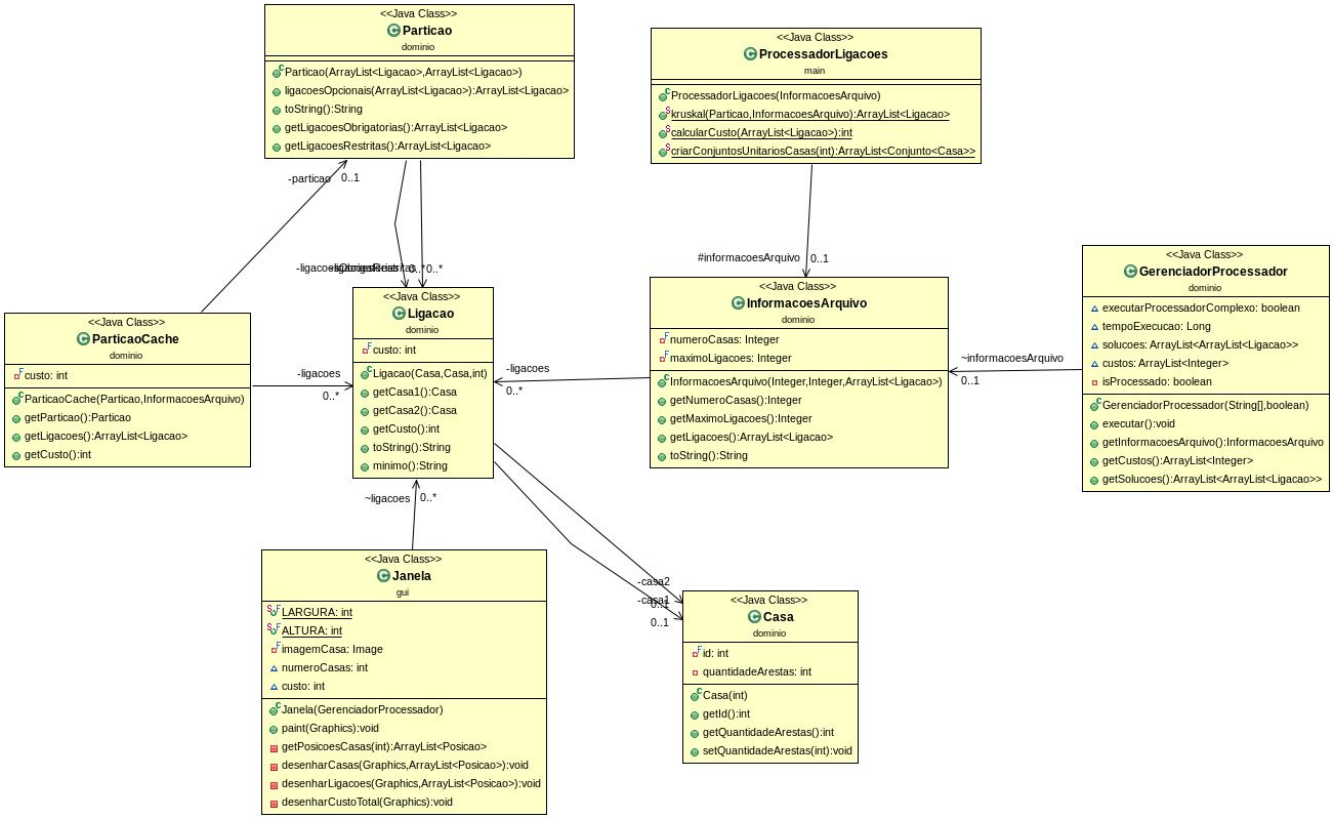


Diagrama de Classes

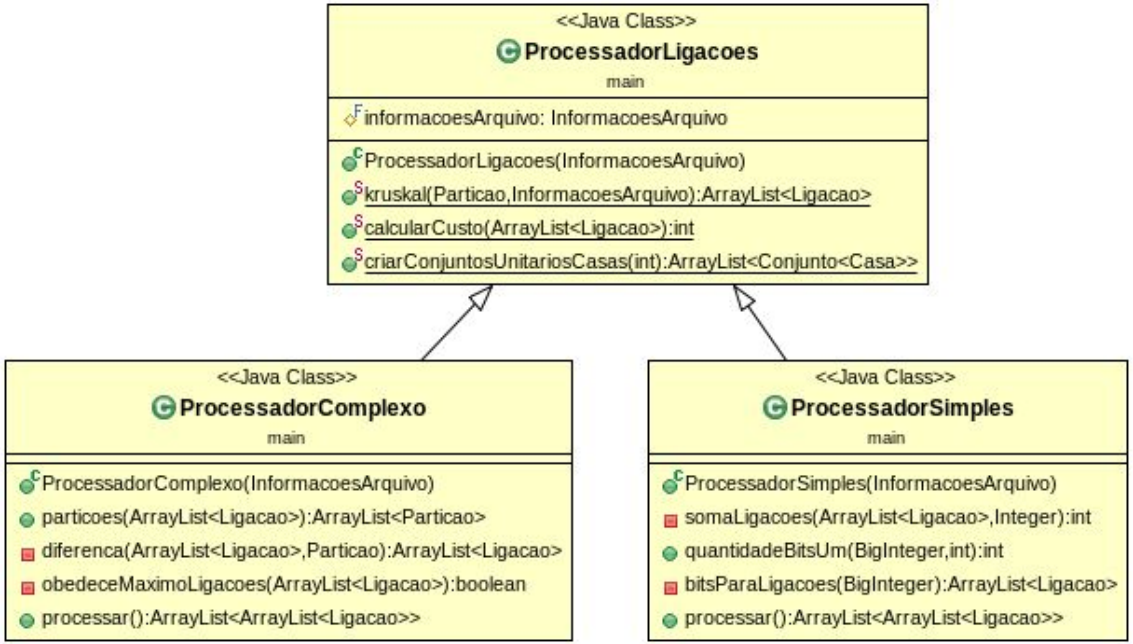
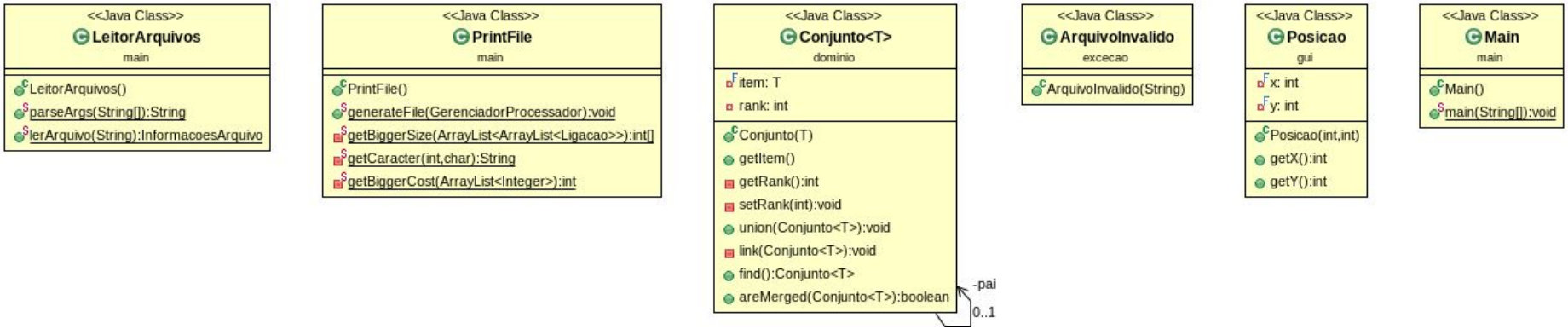


Diagrama de Classes



Etapa 1

Kruskal

```
algorithm Kruskal(G) is
    F:= ∅
    for each v ∈ G.V do
        MAKE-SET(v)
    for each (u, v) in G.E ordered by weight(u, v), increasing do
        if FIND-SET(u) ≠ FIND-SET(v) then
            F:= F ∪ {(u, v)} ∪ {(v, u)}
            UNION(FIND-SET(u), FIND-SET(v))
    return F
```

K-best

List = {A}
Calculate_MST (A)
while MST ≠ ∅ **do**
    Get partition P_s ∈ List that contains the smallest spanning tree
    Write MST of P_s to Output_File
    Remove P_s from List
    Partition(P_s).

PROCEDURE PARTITION (P)
$P_1 = P_2 = P$;
for each edge i in P **do**
if *i not included in P and not excluded from P* **then**
    make i excluded from P_1;
    make i included in P_2;
    Calculate_MST (P_1);
    if Connected (P_1) **then**
        add P_1 to List;
    $P_1 = P_2$;

Etapa 2

Obrigado!