

Appunti di Linguaggi e Traduttori

A cura di:
Francesco Refolli
Matricola 865955

Anno Accademico 2024-2025

Chapter 1

28/02/25

1.1 Strutture Dati

Le strutture dati utilizzate sono gli alberi, i grafi, le pile, le code.

1.2 Algoritmi

1.2.1 Visita Pre

E' un processo di ereditarieta' $\Rightarrow attr(child) = compute(attr(root))$.

```
1 def preorder(tree, visit):
2     root, children = tree
3     visit(root)
4     for child in children:
5         preorder(child, visit)
```

1.2.2 Visita Post

E' un processo di sintesi $\Rightarrow attr(root) = sum([attr(child) for child in children])$.

```
1 def postorder(tree, visit):
2     root, children = tree
3     for child in children:
4         preorder(child, visit)
5     visit(root)
```

1.2.3 Visita Level

```
1 def levelorder(tree, visit):
2     Q = Queue([])
3
4     Q.enqueue(tree)
5     while len(Q) > 0:
6         tree = Q.dequeue()
7         root, children = tree
8         visit(root)
9         for child in children:
10             Q.enqueue(child)
```

1.3 Grafi

Diagramma di ASSE

1.3.1 Depth First

```
1 def depthfirst(adiacency, start, visit):
2     S = Set({})
3     def recursive(src):
4         S.add(src)
5         visit(src)
6         for dst in adiacency[src]:
7             if dst not S:
8                 recursive(dst)
```

1.3.2 Breadth First

```
1 def breadthfirst(adiacency, start, visit):
2     S = Set({})
3     Q = Queue({start})
4     while len(Q) > 0:
5         src = Q.dequeue()
6         S.add(src)
7         visit(src)
8         for dst in adiacency[src]:
9             if dst not S:
10                 Q.enqueue(dst)
```

1.3.3 BackTracking

Abilita' di rendersi conto che il percorso in profondita' non e' ottimale e quindi tornare sui miei passi per processare altri percorsi.

```
1 def backtrack(candidate):
2     if reject(candidate):
3         return
4     if accept(candidate):
5         output(candidate)
6     s = first(candidate)
7     while s:
8         backtrack(s)
9     s = next(candidate)
```