

Mergesort con allocazione singola

Realizzare una versione del merge sort `MergeSort(A,B, ...)` dove A è l'input, mentre B è un array della stessa dimensione utilizzato per la memorizzazione temporanea. La procedura Merge fa alternativamente il merge dei sottoarray in A verso B e vice versa.

Soluzione: L'idea è quella di usare un flag booleano `DestA`, che dica se la destinazione dell'ordinamento è A (quando vero) oppure B (quando falso).

```
# se  destA=true -> ordina  A[p,r] con risultato in A[p,r]
# altrimenti    -> ordina  A[p,r] con risultato in B[p,r]
```

```
MergeSort (A, B, p, r, destA)
    if p < r
        q = (p+r)//2
        MergeSort (A, B, p, q, not destA)
        MergeSort (A, B, q+1, r, not destA)
        if destA
            Merge (B, A, p, q, r)
        else
            Merge (A, B, p, q, r)
    else
        if not destA
            B[p]=A[p]
```

```
# fa il merge di S(ource) in T(arget)
```

```
def Merge (S, T, p, q, r):
    i=p
    j=q+1
    for k = p to r
        if (i <= q) and (not (j <= r) or (S[i] <= S[j])):
            T[k] = S[i]
            i ++
        else:
            T[k] = S[j]
            j ++
    }
```