Esame di Algoritmi e Strutture di Dati I
Mercoledì 30 Luglio 2003

Nome:
Cognome:
Matricola:

- **Scrivere in forma leggibile il proprio nome, cognome e matricola sul testo del compito e su ogni foglio consegnato;**
- **Consegnare solo la bella copia e il testo del compito;**
- **Non è possibile consultare alcun tipo di materiale didattico;**
- **Non è possibile uscire dopo l'inizio dello scritto.**

Esercizio 1 (*Punti 30*)

Una ***tabella a indirizzamento diretto*** *è un vettore* $T[0, \ldots, n]$ *che contiene in posizione* i *un puntatore all'oggetto con chiave* i*, se tale oggetto è presente in tabella, oppure* NIL*, se tale oggetto non è presente in tabella. Scrivere le seguenti procedure e valutarne la complessità pessima:*

1. $Next(T, i)$ *che ritorna la posizione nella tabella* T *in cui si trova l'oggetto con chiave più piccola tra quelli che hanno chiave maggiore di* i*, oppure* NIL *se tale oggetto non esiste;*
2. $Prev(T, i)$ *che ritorna la posizione nella tabella* T *in cui si trova l'oggetto con chiave più grande tra quelli che hanno chiave minore di* i*, oppure* NIL *se tale oggetto non esiste.*

Utilizzando le procedure Next *e* Prev*, scrivere le seguenti procedure valutandone la complessità:*

1. $Max(T)$ *che ritorna la posizione nella tabella* T *in cui si trova l'oggetto con chiave massima tra quelli presenti in tabella, oppure* NIL *se* T *non contiene alcun oggetto;*
2. $Min(T)$ *che ritorna la posizione nella tabella* T *in cui si trova l'oggetto con chiave minima tra quelli presenti in tabella, oppure* NIL *se* T *non contiene alcun oggetto;*

Soluzione

Algoritmo 1 Next(T,i)

Next(T,i)

```
1: j ← i + 1
2: while j ≤ n and T[j] = NIL do
3:     j ← j + 1
4: end while
5: if j ≤ n then
6:     return j
7: else
8:     return NIL
9: end if
```

Algoritmo 2 Prev(T,i)

Prev(T,i)

```
1: j ← i − 1
2: while j ≥ 0 and T[j] = NIL do
3:     j ← j − 1
4: end while
5: if j ≥ 0 then
6:     return j
7: else
8:     return NIL
9: end if
```

Algoritmo 3 Max(T)

Max(T)

1: **return** $Prev(T, n+1)$

Algoritmo 4 Min(T)

Min(T)

1: **return** $Next(T, -1)$

Tutte le procedure hanno complessità pessima $\Theta(n)$.