Relazione: vantaggi di randomLevel()

Alberto Pochettino Federico Pitaccolo

/\*

 \*Funzione probabilistica che decide la size di ogni nuovo nodo. Più è basso il livello più è alta la probabilità

\*/

int random\_level(){

    int lvl = 1;

  // random() returns a random value in [0...1)

  while ( (random() /(double)RAND\_MAX ) < 0.5 && lvl < MAX\_HEIGHT){

    lvl++;

  }

  return lvl;

}

/\*lvl  probabilità

    1           1/2

    2           1/4

    3           1/8

    ..      ...

    n       1/2^n           \*/

Dato un qualsiasi numero n>=1, la probabilità che random\_level fornisca come valore di ritorno quel numero è

In questo modo ci si assicura che l’algoritmo di ricerca sia davvero di complessità O (log n).

Per quanto riguarda la scelta di MAX\_HEIGHT siamo arrivati alla conclusione che deve crescere al crescere del numero di elementi nella skip list al fine di evitare un numero troppo elevato di salti da nodo a nodo nell’algoritmo di ricerca.

Un caso degenere si verifica quando ci sono tanti elementi nodi in cui il campo size è uguale a MAX\_HEIGHT.

Nel caso in cui MAX\_HEIGHT sia impostato a 1 ci troviamo nel caso in cui la skip list non è in condizione di skippare alcun elemento, rendendola quindi una lista normale.

Il caso ottimale si ha quando solo un elemento nodo arriva ad avere il campo size = MAX\_HEIGHT.