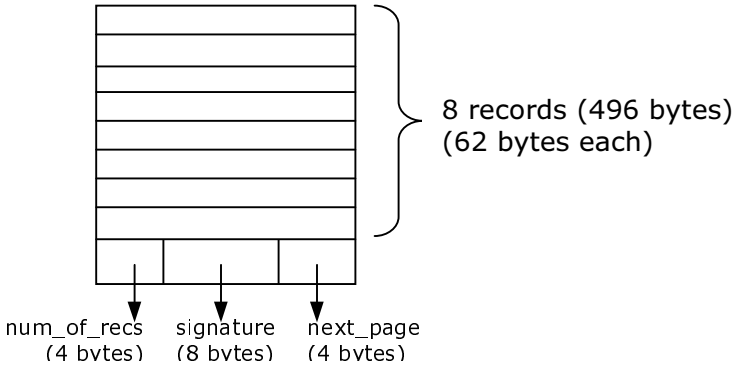
Page Structure



num_of_recs Number of records in page
signature Bloom signature
next_page Next page in chain

Header Information

N Initial number of primary pages (constant)
P Current number of primary pages
O Current number of not empty overflow pages
S Size of primary and overflow page (512 bytes)
L Load factor (70%-80%)
i Current level (initially 0)
R Current number of inserted records
F Address of the first empty overflow page
W Address, where to allocate a new overflow page
next Address of the primary page to split next

Pseudo-code for Insertion

```
Insert(r)       // record r comes from the summary file
                // and has two fields, a for author name and o for data file offset
{
page pp, op;  // primary and overflow page to retrieve

x = hash(r.a);          // integer value representing author name r.a
k = h_i(x) = x % N2^i; // k is the primary page corresponding to record r
if (k < next)   // check if primary page has been split
        k = h_{i+1}(x) = x % N·2^{i+1};     // recalculate k
pp = get_primary_page(k); // retrieve actual primary page
if (pp.num_of_recs < 8) {    // check for any empty slots in primary page
        place r in position (num_of_recs + 1) of page pp
        update pp.num_of_recs
        update signature
```

```
} else {        // check for any empty slots in connected overflow pages
        if (pp.next_page == 0)       // primary page not connected to any overflow pages
                check F for empty overflow pages
                if there is one
                        op = get_overflow_page(F);
                        update pp.next_page
                        update F
                        update O
                else
                        op = allocate_new_page(W);
                        update pp.next_page
                        update W
                        update O
        else
                retrieve next overflow page in the chain till you find a half-full one
                if all overflow pages in the chain are full
                        check F for empty overflow pages
                        .
                        .
                        .       // same as above
                        .
                        .
        // at this point you have an overflow page and you want to insert your record
        place r in position (num_of_recs + 1) of page op
        update op.num_of_recs
        update signature
        update R
}

if ( R > (P + O)·8·L ) {        // check if you have reached the load factor
        // if so, then split the primary page pointed by next
        create a new primary page at address (P + 1)·512
        update P
        // first iteration of the old chain
        rehash all records in the old chain  // you can use insert function recursively
                                             // use h_{i+1}(x) hash function
        // second iteration of the old chain
        pack records in the old chain
        update any overflow pages that became empty and update F and O
        update next
        if (next == N·2^i)      // number of primary pages has been doubled
                next = 0        // next points to the first primary page
                i = i +1        // update level
}
}
```