Quiz 4

Deadline	Friday, 10 April 2020 at 11:59PM
Latest Submission	Monday, 06 April 2020 at 9:45PM
Maximum Mark	4

Question 1 (1 mark)

Consider a table defined as:

```
create table Employees (
    id integer primary key,
    name varchar(50),
    address varchar(100), ...
);
create index on table Employees (id);
```

Employee records are added to the table as employees are hired, and never removed. Thus, the file holding the data pages of the `Employees` table has data pages packed as full as possible, with no overflow pages, and the file is sorted on the `id` attribute. The table has the following characteristics:

B =8192 bytes per page (both for data pages and index pages)
r =10000 total tuples
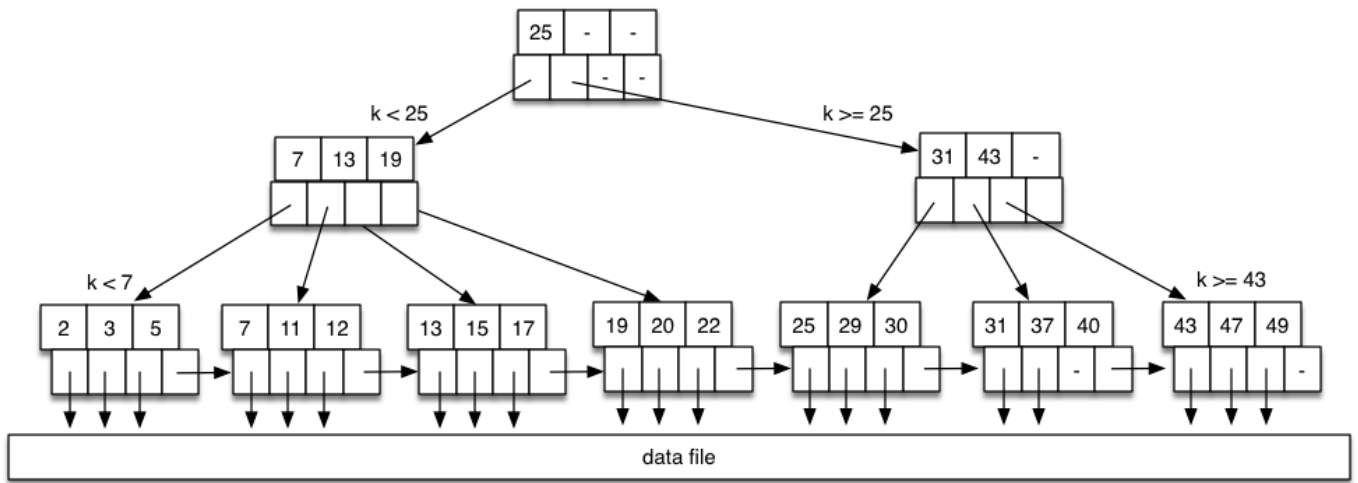R =200 bytes (average, but assume all tuples are this length)
R_i=8 bytes (each index tuple contains (Employee.id,pageID)

Assuming that data pages have a 192-byte header and and index pages have a 32-byte header, and that a sparse index is used, what are the values of b (the total number of data pages) and i (the total number of index pages)? The choices below are presented as *(b,i)* pairs.

(a) ⦿	(250, 1)
(b) ◯	(250, 10)
(c) ◯	(10000, 1)
(d) ◯	(10000, 10)
(e) ◯	None of the other options is correct.

Question 2 (1 mark)

Consider the following B-tree

```
                        25  -  -
               /    -  -  -
          k < 25              k >= 25
      7  13  19                       31  43  -

 k < 7                                         k >= 43
2 3 5   7 11 12   13 15 17   19 20 22   25 29 30   31 37 40   43 47 49

                        data file
```

If the key value 21 is inserted into this tree, what will be the final value(s) in the root node?

If a node needs to be split, assume that the original middle value is the one promoted (e.g. if [2,3,5] was split, then 3 would be promoted).

(a) ●	13 and 25
(b) ○	21 and 25
(c) ○	7 and 25
(d) ○	19 and 25
(e) ○	None of the other options is correct.

Question 3 (1 mark)

Consider a relation R(a,b,c) implemented as a multi-attribute hashed file with the following parameters:

b = 1024 data pages
d = 10 bits for hash values
d_a = 5 bits contributed by a
d_b = 3 bits contributed by b
d_c = 2 bits contributed by c

Assuming that there are no overflow pages, how many pages will be fetched in answering the query:

```
select * from R where a=3 and c=1;
```

(a) ○	3
(b) ●	8
(c) ○	32
(d) ○	1024

| (e) ○ | None of the other options is correct. |

Question 4 (1 mark)

Consider a table defined as

```
create table (
    id      integer primary key,
    name    text,
    colour varchar(8),
    price   float
);
```

Consider now a bitmap index on the values of the `colour` attribute. There are 8 distinct colour values, and r=16384 records, with page size B=8192. The index is arranged as a header page containing just *(key,offset)* pairs, where the *key* is a colour value and the *offset* is the location in the index where the bit-string for that *key* starts. The rest of the pages in the index are used to store bit-strings. If bit-strings stored as compactly as possible in pages (i.e. each index page consists entirely of bit-strings), how many index pages are required? Include the header page.

(a) ○	2
(b) ⦿	3
(c) ○	5
(d) ○	9
(e) ○	None of the other options is correct.

✔ Submit