Alex Stewart

CS350 HW8

File Structures, Indexing, Hashing (DUE Week 9 - Friday night at midnight)

17.41

a.

```
*start_location = 200;
int a = 5;
int b = 2;
int record_size = 25;
for (int byte = 0; byte >=25; byte++)
{
        while(byte)
        {
                a = start_location +(record_size*a)+b;
        }
}
```

-The code above assumes that the starting location of a memory address is 200. Computer memory records are stored in a block. Block size is shown by byte bytes and record size is shown by record_size.

-This code sets memory address to 200 and then records are stored in blocks where size is “byte bytes” and record size is “record_size”

b.

```
*start_location = 200;

int a = 5;

int b = 2;

int record_size = 25;

int i = 0;

int block_size;

int field_size = 1;

for (int byte = 0; byte >=25; byte++)

{

          while($)

          {

                    current_location = current_location + 25byte;

                    while(byte)

                    {

                              i = i + 2*(field_size+1)

                    }

          }

}
```

-$ is a separator character. loop while there is a separator,

update current location and i while you loop

c.

```
*start_location = 200;

int a = 5;

int b = 2;

int record_size = 25;

int field_size = 1;

int cur_record;

boolean empty = ReadFirstByte(field_size);

if (!empty)

{

        cur_record += field_size.length();

 }

else if(cur_record!=record_size)

{

        empty=false;

}

else

{

        record.push_back(*this);

}
```

- each record has an end of record byte. Then access records by Moving byte by byte.

then check value of specified condition.

d.

```
if(!empty)
{
        cur_record += field_size.length();
}
```

-no record length required because of the multiple block records

e.

```
if(cur_record!=record_size)
{
        empty=false;
}
```

-record length can be skipped due to optional fields

f.

```
if(cur_record > record_size)
{
        record.push_back(*this);
}
```

-due to varying record size there is no record greater than record_size

18.19 on next page

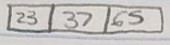
Alex Stewart HW 8

CS 350

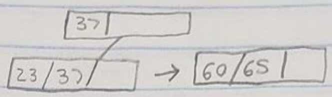
18.19

A. 15 keys : 23, 65, 37, 60, 46, 92, 48, 71, 56, 59, 18, 21, 10, 74, 78

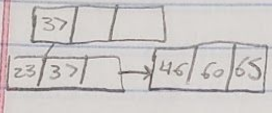
1. Insert 23,37,65



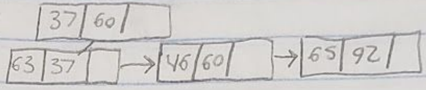
2. Insert 60



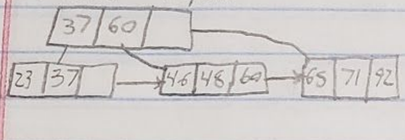
3. Insert 46



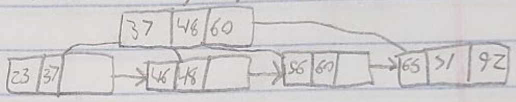
4. Insert 92



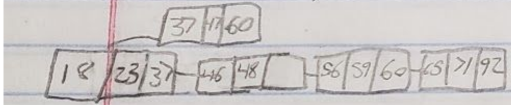
5. Insert 48, 71



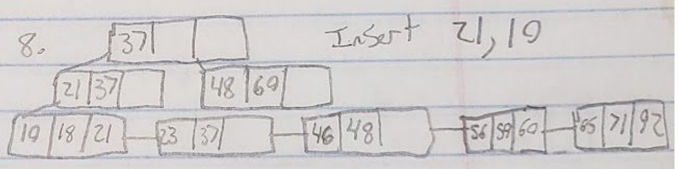
6. Insert 56, 46, 48, 56, 60



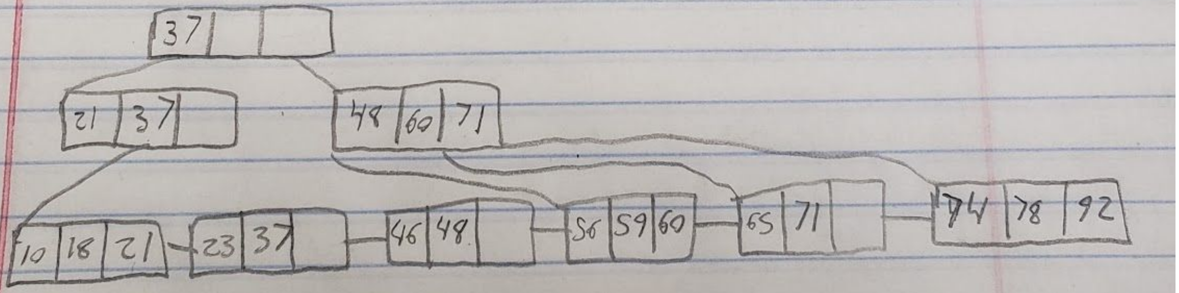
7. Insert 59, 18



8. Insert 21, 10



9. Insert 74 then 78


37
21 | 37
48 | 60 | 71
10 | 18 | 21
23 | 37
46 | 48
56 | 59 | 60
65 | 71
74 | 78 | 92