個人資訊

姓名：曾益銘
學號：408410081
Email: yikming2222@gmail.com

程式資訊

程式語言：Java
版本：openjdk 20, javac 20, java 20
Note:

- 還測試過 openjdk 11, javac 11, java 11，其他 java 版本不確定是否能夠編譯執行
- 如果測試作業系統是 mac 的話不要用 cmd + q 把 GUI 視窗關閉，否則整個程式會關閉。此外，如果在 mac 把 GUI 視窗縮放到最大，出現以下錯誤可以忽略，該錯誤不影響程式操作（該錯誤只有會在 mac 出現）

```
1   HIToolbox                       0x00000001b0c845c8 _ZN15MenuBarInstance22EnsureAutoShowObserverEv + 120
2   HIToolbox                       0x00000001b0c84188 _ZN15MenuBarInstance14EnableAutoShowEv + 60
3   HIToolbox                       0x00000001b0bf18bc _ZN15MenuBarInstance21UpdateAggregateUIModeE21MenuBarAnimationStylehhh + 1184
4   HIToolbox                       0x00000001b0c84004 _ZN15MenuBarInstance19SetFullScreenUIModeEjj + 180
5   AppKit                          0x00000001aaa73d30 -[NSApplication _setPresentationOptions:instance:flags:] + 956
6   AppKit                          0x00000001aa90993c -[NSApplication _updateFullScreenPresentationOptionsForInstance:] + 404
7   CoreFoundation                  0x00000001a7526560 __CFNOTIFICATIONCENTER_IS_CALLING_OUT_TO_AN_OBSERVER__ + 148
8   CoreFoundation                  0x00000001a75c4044 ___CFXRegistrationPost_block_invoke + 88
9   CoreFoundation                  0x00000001a75c3f8c _CFXRegistrationPost + 440
10  CoreFoundation                  0x00000001a74f7b64 _CFXNotificationPost + 708
11  Foundation                      0x00000001a83e738c -[NSNotificationCenter postNotificationName:object:userInfo:] + 88
12  AppKit                          0x00000001aaa742b4 spacesNotificationHandler + 96
13  SkyLight                        0x00000001ac161214 _ZN12_GLOBAL__N_123notify_datagram_handlerEj15CGSDatagramTypePvmS1_ + 896
14  SkyLight                        0x00000001ac48d4d4 _ZN21CGSDatagramReadStream26dispatchMainQueueDatagramsEv + 228
15  SkyLight                        0x00000001ac48d3d0 ___ZN21CGSDatagramReadStream15mainQueueWakeupEv_block_invoke + 28
16  libdispatch.dylib               0x00000001a72c89dc _dispatch_call_block_and_release + 32
17  libdispatch.dylib               0x00000001a72ca504 _dispatch_client_callout + 20
18  libdispatch.dylib               0x00000001a72d8d1c _dispatch_main_queue_drain + 928
19  libdispatch.dylib               0x00000001a72d896c _dispatch_main_queue_callback_4CF + 44
20  CoreFoundation                  0x00000001a7572d40 __CFRUNLOOP_IS_SERVICING_THE_MAIN_DISPATCH_QUEUE__ + 16
21  CoreFoundation                  0x00000001a75307c0 __CFRunLoopRun + 2036
```

- 最好是在 mac 或 unix-like 系統執行 ，因為 windows 的 cmd 有 ANSI color code 的問題無法顯示顏色，而用 ANSI code 代替顯示，導致 cmd 版面較亂（如下圖，例如 <-[36;1m 本應是顯示顏色）

```
←[36;1mWelcome!←[0m
←[35;1m
←[31;1m(enter h to show manual or q to quit)←[0m
please enter a num to set the order of b+tree:
←[0m
2
←[35;1m
←[31;1m(enter c to change order of b+tree or h to show manual or q to quit)←[0m
please select an operation on b+tree:
(1)insert (2)lookup (3)display (4)display(GUI)
```

程式講解與操作

講解

1. 直接執行 ./run.sh，note: 確保程式執行的目錄具備 executable 權限

Note: run.sh 只能在 macOS 和 unix-like 系統使用，windows 的話需要重新編譯 Main.java (編譯指令： javac Main.java)，接著再執行（執行指令： java Main）

2. 互動式模式中，如果輸入錯誤會有對應的提示，然後按照提示修改輸入即可
3. 模式 state 可分為：
   - 輸入 B+tree order state
   - 操作 B+tree state，B+tree 有 insert 、search(包含 range search)、display 、display(GUI) 功能
   - Note: 在每個模式都有額外的指令，如圖1 的紅色文字說明，“enter c to change order of b+tree or h to show manual or q to quit”，輸入 c 可以回到上一個 state（如圖2），而不用重新執行程式才能修改 B+tree order，輸入 q 則會結束程式而不用自己輸入 EOF 或 ctrl-C signal INT 來結束程式（如圖3）

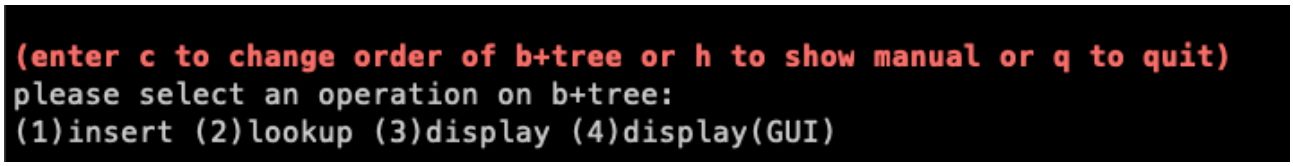

圖1

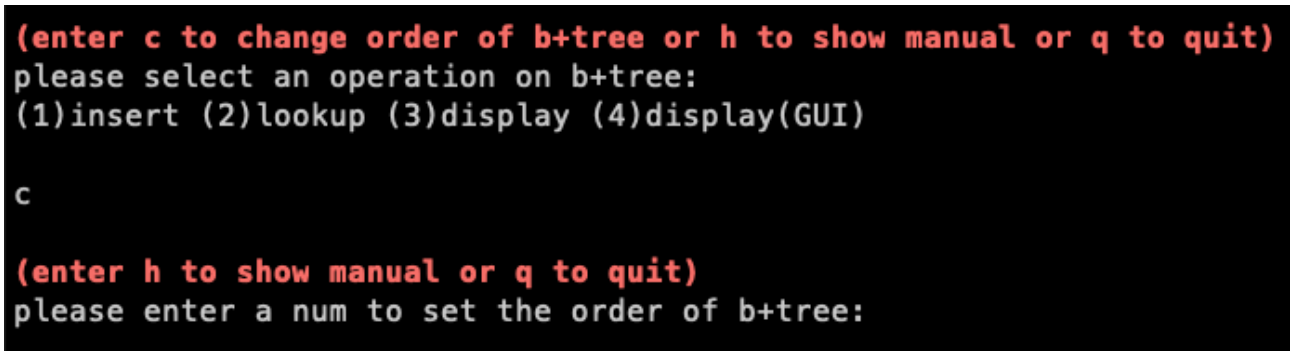

圖2

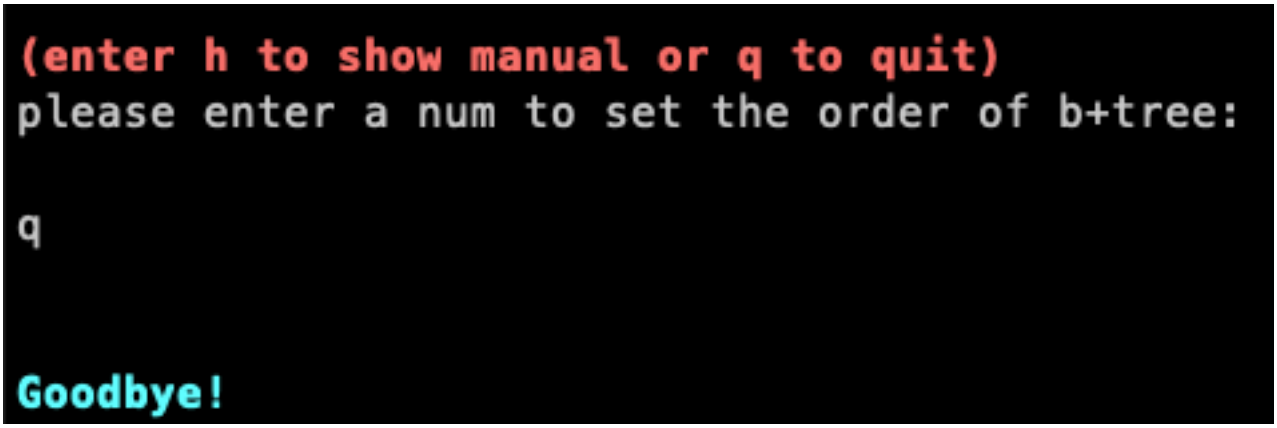

圖3

操作

1. ./run.sh

2. 輸入 order ，這裡示範輸入 2（如圖4）

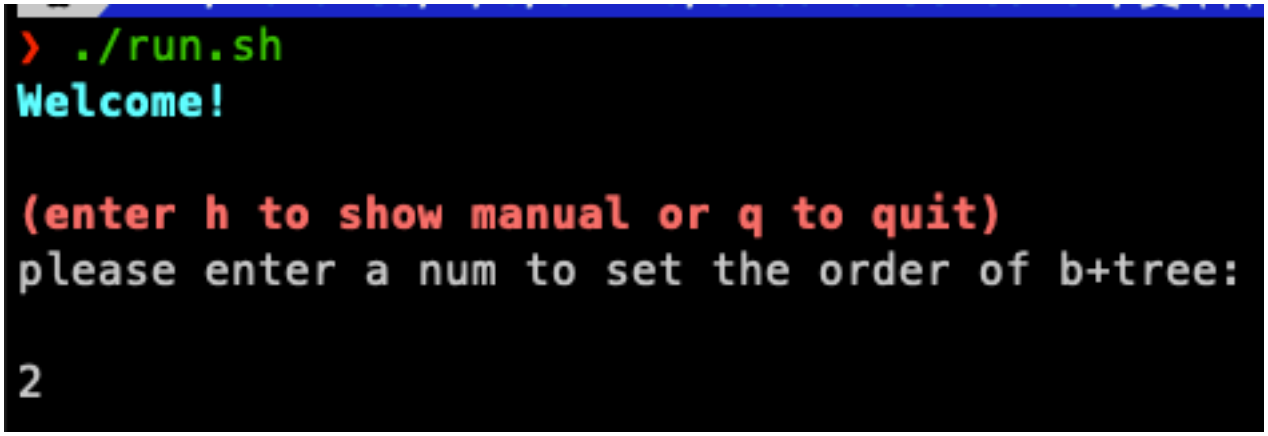

圖4

3. 輸入 1 進入 insert key state（如圖5），可以選擇一次輸入多筆 key （用空格隔開，如圖6），也可以選擇輸入單筆 key（如圖7），insert 結束後會回到選擇操作 B+tree state，這時候可以選擇其他的操作

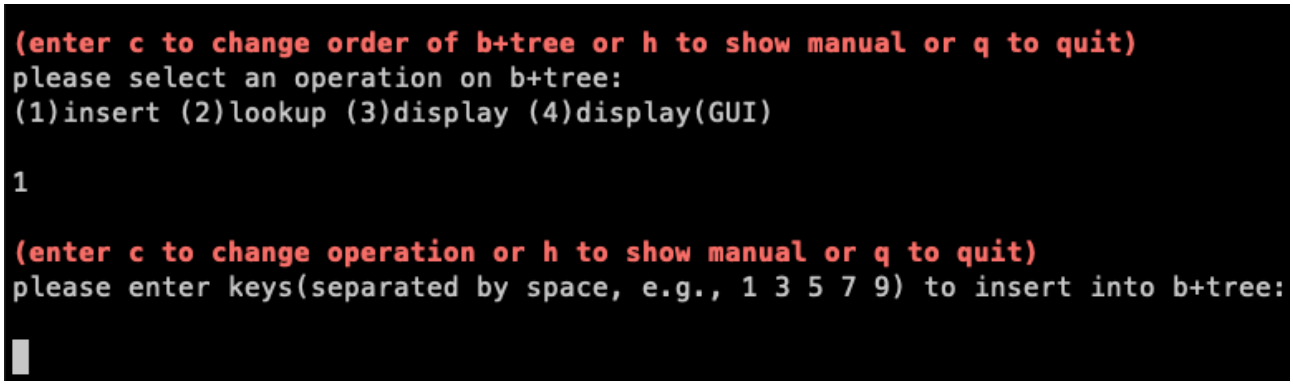

圖5

```
(enter c to change operation or h to show manual or q to quit)
please enter keys(separated by space, e.g., 1 3 5 7 9) to insert into b+tree:

1 2 3 4 50 100
Insert key 1:
[1,_,_,_]

Insert key 2:
[1,2,_,_]

Insert key 3:
[1,2,3,_]

Insert key 4:
[1,2,3,4]

Insert key 50:
(3:_:_:_)
  [1,2,_,_]
  [3,4,50,_]

Insert key 100:
(3:_:_:_)
  [1,2,_,_]
  [3,4,50,100]
```

圖6

```
(enter c to change operation or h to show manual or q to quit)
please enter keys(separated by space, e.g., 1 3 5 7 9) to insert into b+tree:

100
Insert key 100:
(3:_:_:_)
  [1,2,_,_]
  [3,4,50,100]
```

圖7

4. 輸入 2，進入 search 模式（如圖8），支援單筆 search 和 range search 。如果要用單筆 search 的話，需要輸入格式為 「= x」（如圖9）。如果要使用 range search 的話，需要輸入格式為 「op a」（如圖10）和「a op key op b」（如圖11），op = [<, <=, >, >=]，a 、b = integer，search 結束後會回到選擇操作 B+tree state，這時候可以選擇其他的操作。
   Note: 若有 duplicated keys 會一起搜尋

```
(enter c to change order of b+tree or h to show manual or q to quit)
please select an operation on b+tree:
(1)insert (2)lookup (3)display (4)display(GUI)

2

(enter c to change operation or h to show manual or q to quit)
please enter an expression for range searching in b+tree:
for example: = 6, > 6, >= 6, < 6, >= 6, 32 <= key < 100, 22 < key < 101
```

圖8

```
(enter c to change operation or h to show manual or q to quit)
please enter an expression for range searching in b+tree:
for example: = 6, > 6, >= 6, < 6, >= 6, 32 <= key < 100, 22 < key < 101

= 100

Result: 100

Duplicated keys: {100}
```

圖9

```
(enter c to change operation or h to show manual or q to quit)
please enter an expression for range searching in b+tree:
for example: = 6, > 6, >= 6, < 6, >= 6, 32 <= key < 100, 22 < key < 101

< 100

Result: 3,4,50,1,2

No duplicated keys!
```

圖10

```
(enter c to change operation or h to show manual or q to quit)
please enter an expression for range searching in b+tree:
for example: = 6, > 6, >= 6, < 6, >= 6, 32 <= key < 100, 22 < key < 101

4  <= key <= 100

Result: 4,50,100

Duplicated keys: {100}
```

圖11

5. 輸入 3，顯示當前 B+tree 的 pre-order 樣子（如圖12）

```
(enter c to change order of b+tree or h to show manual or q to quit)
please select an operation on b+tree:
(1)insert (2)lookup (3)display (4)display(GUI)

3
(3:_:_:_)
  [1,2,_,_]
  [3,4,50,100]
```

圖12

6. 輸入 4，GUI 方式顯示當前 B+tree 的 pre-order 樣子（如圖13）

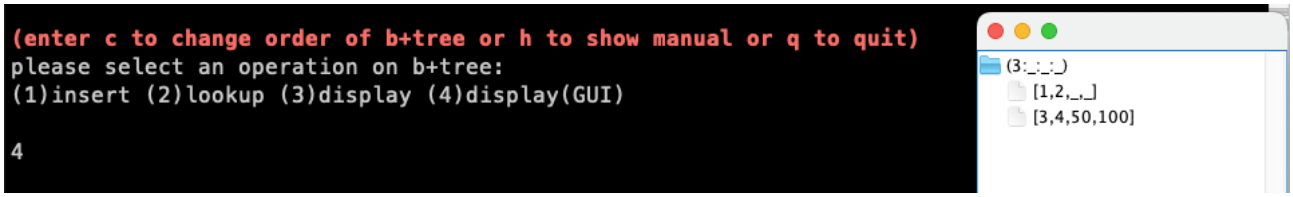

圖13

Note：若某些功能不清楚使用方式或不同 state 的合法輸入是什麼的話，可以輸入 h 印出 user manual 查詢（如圖14）

```
(enter c to change order of b+tree or h to show manual or q to quit)
please select an operation on b+tree:
(1)insert (2)lookup (3)display (4)display(GUI)

h
Usage:
        (1) When you are entering the order of b+tree, must be >= 1, e.g, 1, 2, 3, ...
        (2) When you are selecting the operation of b+tree, valid operation are 1, 2, 3 and 4 only
                1 is insert
                2 is single search or range search depends on the expression
                3 is display
                4 is display in GUI
        (3) When you are entering the keys to insert into b+tree,
                it could be one or more keys, note that multiple keysshould separated by space, e.g., 32, 64 23 12 18 66 70 80
        (4) When you are entering the expression to search from b+tree,
                it could be single search or range search, the format as follow:
                        - single search, = x, x = {integer}, e.g., = 23
                        - range search, op x or a op key op b, x and a and b = {integer}, op = {<, <=, >, >=}, e.g., >= 32, < 32, 60 <= key <= 100
```

圖14

B+tree 功能講解

- Insert 當發生 overflow 的時候 leaf node 會 split，然後取 middle key copy up 當作新的 index ，而如果 overflow 發生在 internal node 的話 internal node 會 split 之後會取 middle key push up 當作新的 index，因爲 insert 是遞迴函數所以會一直檢查是否 overflow 直到 root
- Search 分為：
  - 單筆 search，和傳統的 B 樹搜尋方式一樣，一直走訪到 leaf node，最後比較 leaf node 裡面的 key 是否有匹配的 key
  - Range search，B+tree 的 leaf node 是用 doubly linked list 串在一起的，因此當找到一個 key 的起點後，往右就是 >= key，往左就是 < key
- Duplicated key 處理：雖然說 B+tree 不應該有 overflow page 因為會降低效能（就如 ISAM），但這裡選擇用 overflow page 的方式來處理 duplicated key 因為實作起來較簡易，加上也能夠解決此問題。有時候 trade-off 是需要的。
- display 圖形化顯示 B+tree 是使用 Swing library 內建的 Tree 物件建構的

非自行完成部分

1. ConsoleColors class 是直接使用 stack overflow 別人寫好的(此物件包含的變數或函數為"非"作業主幹)，資源網址：https://stackoverflow.com/questions/5762491/how-to-print-color-in-console-using-system-out-println
2. Swing GUI 參考("非"作業主幹), https://www.tutorialspoint.com/how-to-expand-jtree-row-to-display-all-the-nodes-and-child-nodes-in-java 、https://www.javatpoint.com/java-jtree
3. 如何從 order set 取出最後一個元素("非"作業主幹)，https://stackoverflow.com/questions/14024022/getting-last-of-linkedhashset
4. Duplicated key 參考("非"作業主幹), https://thodrek.github.io/cs564-fall17/lectures/lecture-13/Lecture_13_Btree.pdf

Bonus

1. 互動式程式執行模式
2. display 圖形化顯示
3. Search and range search
4. Duplicated key 處理
5. 程式 User manual