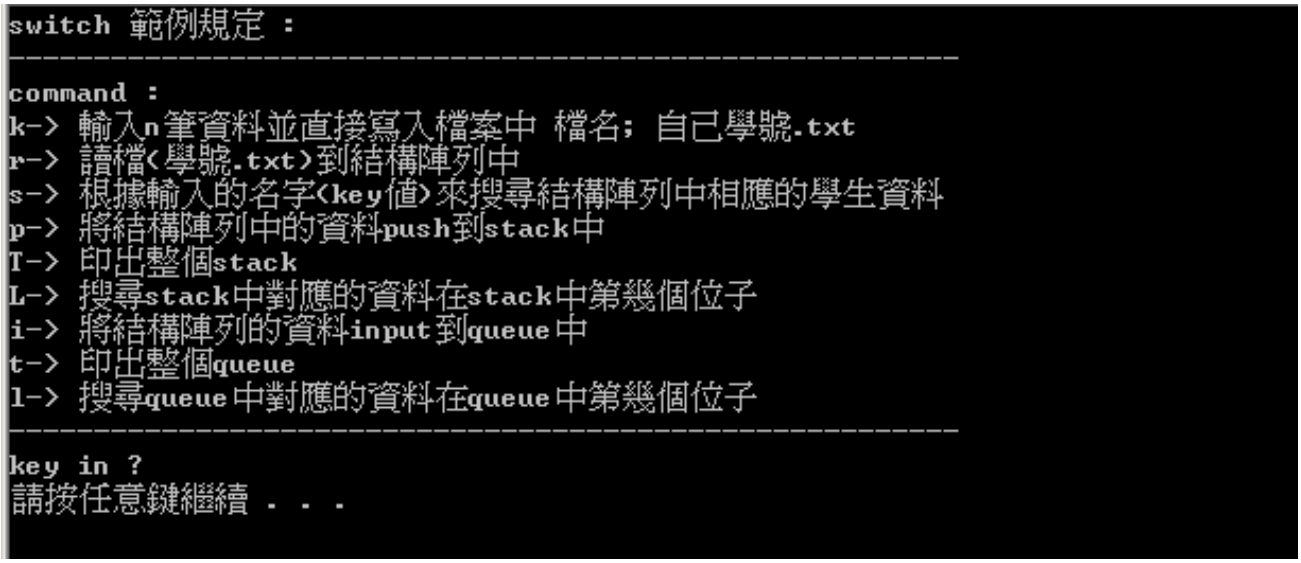
National Sun Yat-sen University Department of Computer Science and Engineering
C programming Final-Term

Note：

學號.txt 檔中每一題結果輸出皆需空一行並標明題號來區隔
請依照題目所提供的函式名稱來撰寫

1. (20%) 請設計一個 switch 選單如下圖，上述功能需在接下來 9 題中逐一完成。



2. (20%) 請在畫面中依照以下格式輸入數筆資料，並將其以同樣格式輸出在**學號.txt** 檔之中

輸入資料格式：

資料數目
N //student_number

學號	**姓名**	**成績**	**年齡**	**月份**	**身高**
M123456789	John	75	19	1	180.0
M987654312	HMD	60	30	5	175.8
……					

Note：每筆資料項皆不超過 20 個字元

函式：int student_data(char *filename);

return N //student_number

3. (20%) 請將第一題所輸出的學號..txt 檔，依同樣格式讀入儲存在 Structure array 之中，在螢幕中印出結果。

```
struct StudentData
{
    char ID[20];       //學號
    char Name[20]    // English name
    int grade;         //成績
    int age;           //年齡
    float tall      // 身高
};
struct StudentData data[N];
```

函式： void ReadFromFile(StudentData *student, char *filename, int studentnum)

Note：每一筆輸出皆需換行

4. (20%) 輸入一個關鍵字(char English Name)，在 **struct StudentData** 陣列中搜尋到相應的學生資料，並將此學生的資料印出在螢幕上。

函式：

```
int array_search (char Name[20]);
    return index;        // 回傳資料在陣列中的位子
```

輸出資料格式：

index: 1
學號: M123456789
姓名 John
成績: 75
年齡: 19
身高: 180

Note：每一筆輸出皆需換行

5. (20%) 將 struct StudentData 陣列，每一筆依序 push 到 Stack 中。
(備註 ：stack push 的方式參考附錄一)

```
struct StudentStackNode
{
   struct StudentData studata;
   struct StudentStackNode * StuLink;
};
```

函式：

```
void PushStudentData (struct StudentStackNode CurrentData,
                      struct StudentStackNode * StuDataPtr);
```

6. (20%)由 Stack pointer 所指的位子開始拜訪所有的節點，讀出所有 Stack 中的內容，並且顯示 struct StudentData 中的所有元素在螢幕上。

範例 ： 如果依序 push 1.2.3.4.5 五筆資料進去 stack 則 traver 的結果爲
5->4->3->2->1

函式：

```
void traver_stack (struct StudentStackNode * StuDataPtr);
```

輸出資料格式：

```
Start                                          // 每次輸出前都加start 區別
M987654312   HMD   60    30    5    175.8   /student2
M123456789   John  75    19    1    180.0   /student1
```

7. (20%)由鍵盤輸入學號並且到 Stack 中去查詢是否有同樣學號的資料，如果有則回傳資料在 Stack 中的位子，並印出在螢幕上 (數字由 1 開始(pointer 所指的位子)，如果沒有找到則回傳 0)。

函式：

```
int search_stack (char ID[20],   struct StudentStackNode * StuDataPtr);
```

Note：char * 爲輸入學號

8. (20%)將 struct StudentData 陣列，每一筆依序 input 到 Queue 中。
struct StudentData (參考第四題)。
(備註 ：Queue insert 的方式參考附錄二)
函式：
void InputStudentData (struct StudentStackNode CurrentData,
struct StudentStackNode * StuDataPtr);

9. (20%)由 Queue pointer 開始拜訪所有 Queue 中的內容，並且顯示
struct StudentData 中的所有元素在銀幕上。

範例 ： 如果依序 input 1.2.3.4.5 五筆資料進去 stack 則 traver 的結果爲
1->2->3->4->5
函式：
void traver_queue (node_pointer);

輸出資料格式：

```
Start                                                  // 每次輸出前都加start 區別
M123456789    John    75    19    1    180.0    /student1
M987654312    HMD     60    30    5    175.8    /student2
```

10 . (20%)由鍵盤輸入學號並且到Queue中去查詢是否有同樣學號的資料，如果有則回傳資料在Queue中的位子 (數字由1開始(pointer 所指的位子)，如果沒有找到則回傳0)。
函式：
int search_queue (char * , struct StudentStackNode * StuDataPtr);
Note：char * 爲輸入學號