

```

//
// main.cpp
// Stack_operation_C++
//
// Created by KHH on 22/9/2018.
// Copyright © 2018年 關曉康. All rights reserved.
//
#include<iostream>
#include<string>
using namespace std;

int global_index_for_top_of_stack = -1;
int index_for_stack_storing_min = -1;

void push_stack(int stack[], int y, int stack_for_storing_min[]);
int pop_stack(int stack[]);
int top_stack(int stack[], int stack_for_storing_min[]);
int FindMin(int stack_for_storing_min[]);

//I assume the length of the stack is 10.
int main(){
    int stack[10] = {0,0,0,0,0,0,0,0,0,0};
    int stack_for_storing_min[10];
    string command;
    int x;
    cout << "Please input if you want the stack the do anything, push/pop/top/find_min and end the input by
    typing 'end'\n";
    do{
        cin >> command;
        if(command=="end"){
            break;
        }
        if(command=="push"){
            cout << "The value that you want to push:";
            cin >> x;
            push_stack(stack, x, stack_for_storing_min);
        }else if(command=="pop"){
            pop_stack(stack);
        }else if(command=="top"){
            cout << top_stack(stack, stack_for_storing_min) << endl;
        }else if(command=="find_min"){
            cout << FindMin(stack_for_storing_min) << endl;
        }
    }while(command!="end");

    cout << "The minimum number in the stack is " << FindMin(stack_for_storing_min) << endl;
    cout << "\nThe min stack now contains the following\n";
    for(int count = 0; count <= index_for_stack_storing_min; count++){
        cout << stack_for_storing_min[count] << " ";
    }

    cout << "\nThe stack now contains the following\n";
    for(int count = 0; count <= global_index_for_top_of_stack; count++){
        cout << stack[count] << " ";
    }
    cout << endl;
    return 0;
}

```

```

void push_stack(int stack[], int y, int stack_for_storing_min[]){
    stack[global_index_for_top_of_stack+1] = y;
    global_index_for_top_of_stack ++;
    if(global_index_for_top_of_stack == 0){
        index_for_stack_storing_min ++;
        stack_for_storing_min[index_for_stack_storing_min] = y;
    }else{
        if(stack_for_storing_min[index_for_stack_storing_min] > y){
            index_for_stack_storing_min++;
            stack_for_storing_min[index_for_stack_storing_min] = y;
        }else{
            index_for_stack_storing_min++;
            stack_for_storing_min[index_for_stack_storing_min] = stack_for_storing_min
            [index_for_stack_storing_min-1];
        }
    }
}

int pop_stack(int stack[]){
    int x = stack[global_index_for_top_of_stack];
    global_index_for_top_of_stack --;
    index_for_stack_storing_min --;
    return x;
}

int top_stack(int stack[], int stack_for_storing_min[]){
    int temp = pop_stack(stack);
    push_stack(stack, temp, stack_for_storing_min);
    return temp;
}

int FindMin(int stack_for_storing_min[]){
    return stack_for_storing_min[index_for_stack_storing_min];
}

```

```
[KHHs-MacBook-Air:Stack_operation_C++ KHH$ g++ main.cpp -o main.o
[KHHs-MacBook-Air:Stack_operation_C++ KHH$ ./main.o
Please input if you want the stack the do anything, push/pop/top/find_min and en
d the input by typing 'end'
push
The value that you want to push:2
push
The value that you want to push:5
push
The value that you want to push:5
push
The value that you want to push:-3
push
The value that you want to push:-3
push
The value that you want to push:20
find_min
-3
pop
pop
pop
find_min
2
end
The minimum number in the stack is 2

The min stack now contains the following
2 2 2
The stack now contains the following
2 5 5
KHHs-MacBook-Air:Stack_operation_C++ KHH$
```