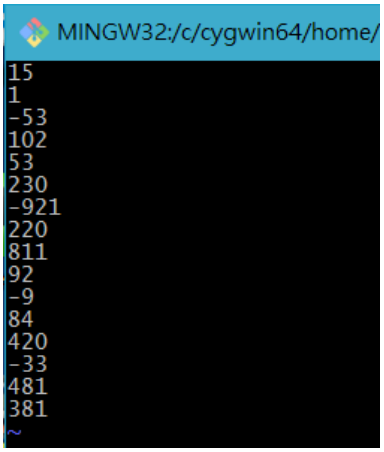
Practice 5 (2016/10/18)
Sorting and Searching

1. File input and output



A. Text file

- ☞ A file is a set of information that is stored in a pre-defined format.
- ☞ A text file is to store information that is encoded by ASCII character set.
- ☞ Users can define their own text file format. For instance, we can say that the number in the first line implies the total number of integers in the file, and then all integers follow the first line.

B. Basic functions to read and print a text file

- ☞ file pointer, fopen, fscanf;

2. Sorting is to arrange a series of numbers in a specified order

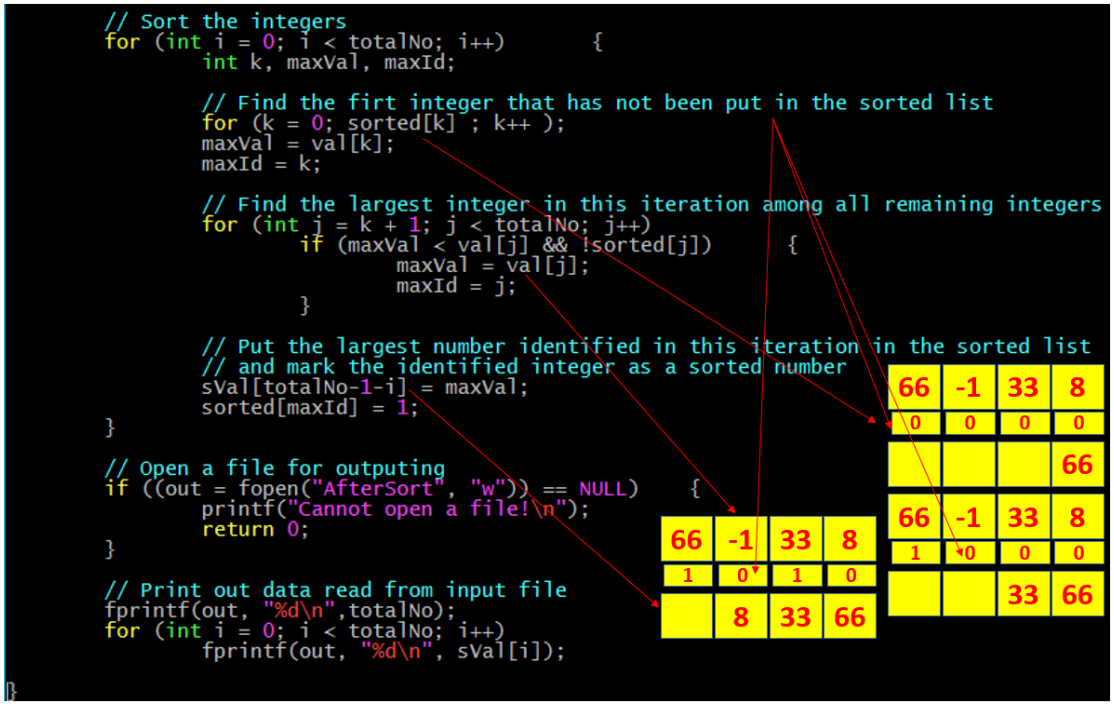
```c
#include <stdio.h>

#define MAX_NO  20000

int main(void)
{
        int totalNo, val[MAX_NO], sVal[MAX_NO], sorted[MAX_NO] = {0};
        FILE *in, *out;

        if ((in = fopen("Set0","r")) == NULL)   {
                printf("Input file does not exist!.\n");
                return 0;
        }

        // Read the total number of integers in the input file
        fscanf(in, "%d", &totalNo);

        for (int i = 0; i < totalNo; i++)       {
                fscanf(in, "%d", &val[i]);
        }

        // Open a file for outputing
        if ((out = fopen("BeforeSort", "w")) == NULL)   {
                printf("Cannot open a file!\n");
                return 0;
        }

        // Print out data read from input file
        fprintf(out, "%d\n",totalNo);
        for (int i = 0; i < totalNo; i++)
                fprintf(out, "%d\n", val[i]);
```

File name

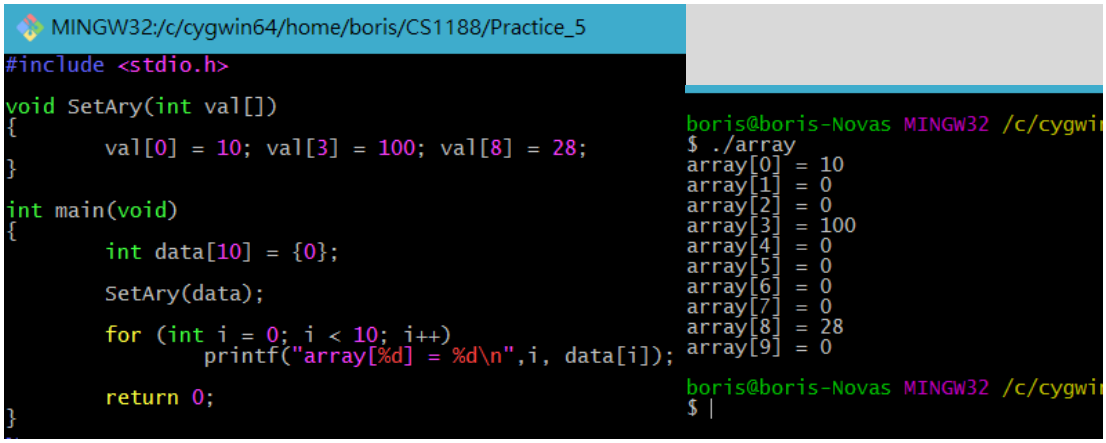
Read mode

File pointer

```
15
1
-53
102
53
230
-921
220
811
92
-9
84
420
-33
481
381
~
```



3. Use array as parameters of function call



4. Problem 2 (120 pts to complete P1 and P2)

Last time we have already realized how to randomly generate different numbers in a row. Today, we practice doing sort and search in an array. (A). At the first of the program, the user is asked if the user wants to generate a set of random integers or read a set of integers from a file. (B-1). If the user wants to generate a set of random integers, ask the user how many integers the user wants to generate and then enter a loop to generate all random numbers one by one. You can set the upper bound of each random integer as 10000 for your convenient use of '%' operator. (B-2). If the user wants to read integers from a file, ask the user to input the file name and then read all integers from the input file. The file format is as follows. The first line is the total number of integers in this file. After this each integer is printed in one line.

(C). After reading integers, perform bubble sorting to sort all integers in increasing

order. (D). Finally enter a while-and-switch structure to as the user which item the user wants to select: (i) to print integers in increasing order in a file named as “IncSort”; (d) to print integers in decreasing order in a file named as “DecSort”; (q) to quit the program. If the user selects the other choice, the program echoes the command choice and asks the user to input again. (100 pts for these features)

```
$ ./sortNum2
Please select the way to read a set of integers.
1: random numbers; 2: read integers from a text file.
1
How many number of integers do you want to input?
10
10 random integers have been generated.
Specify the operation you want to do.
i: print integers in increasing order;
d: print integers in decreasing order.
b: perform a binary search.
q: Quit the program.
i
Specify the operation you want to do.
i: print integers in increasing order;
d: print integers in decreasing order.
b: perform a binary search.
q: Quit the program.
r
Input error!
Specify the operation you want to do.
i: print integers in increasing order;
d: print integers in decreasing order.
b: perform a binary search.
q: Quit the program.
p
Input error!
Specify the operation you want to do.
i: print integers in increasing order;
d: print integers in decreasing order.
b: perform a binary search.
q: Quit the program.
```

```
$ ./sortNum2
Please select the way to read a set of integers.
1: random numbers; 2: read integers from a text file.
2
Please specify the filename you want to open and read.
Set1
All integers in file Set1 have been read.
Specify the operation you want to do.
i: print integers in increasing order;
d: print integers in decreasing order.
b: perform a binary search.
q: Quit the program.
d
Specify the operation you want to do.
i: print integers in increasing order;
d: print integers in decreasing order.
b: perform a binary search.
q: Quit the program.
q

boris@boris-Novas MINGW32 /c/cygwin64/home/boris/CS1188/I$ cat DecSort
$ cat Set1                                                 9
9                                                          9664
9664                                                       8393
5423                                                       7217
7217                                                       5768
5768                                                       5423
2461                                                       5272
5272                                                       2668
1676                                                       2461
2668                                                       1676
8393
```

Features for additional 20 points (120 pts for all features of this practice)

When the sorting is done, it is well prepared to work on the binary search. The user inputs a number and then the program identifies the array location of the number. If the number is not in the array, prompt a message. If the number is in the array, tell the user the array index of the number.