# CS29002 Algorithms Laboratory

## Assignment No: 1

Last date of submission: 27–July–2016

---

The Department of Symbiartic Engineering offers both arts and science courses. You have a list of students entering the second year. Each student record consists of a nine-letter roll number (like 15SY10049, 15SY60D04, 15SY92P15), an arts marks (out of 100, an integer), and a science marks (again an integer, out of 100). A student record typically contains many other fields, but you do not need to worry about those for the sake of this assignment. Assume that the list is sorted with respect to the roll numbers.

An array of structures stores the list. According to the policy of the Department, the list is presented to you in a read-only array, and you are not allowed to modify the array or copy any record from the array to any other location in the memory. You may only visit any record, read any field, do comparisons on this value, and if needed print it. You are forbidden to use any record or any of its components on the right side of any assignment statement. You are, however, allowed to use your own arrays for storing other information. More precisely, you are allowed to use $O(n)$ additional space in your functions, where $n$ denotes the number of students in the Department.

**Part 1:** Handle queries of the form $(sub, a, b)$, where $sub$ is the subject (Arts/Science) and $a, b$ are two integers satisfying $0 \leqslant a \leqslant b \leqslant 100$. The query should print all the student roll numbers and their marks in the specified subject provided that the marks are in the range $[a, b]$. The printing should be in the sorted order of the marks (higher to lower). Notice that some marks may appear multiple times, and some marks may be missing. All students with marks in the query interval $[a, b]$ should be reported.

Assume that the number of such queries to be made is $\gg n^2$. You should achieve a worst-case running time of $O(t + \log n)$ per query, where $t$ is the number of records printed in the query.

**Part 2:** After the results of the first year are out, the Head of the Department needs to decide for each student whether (s)he will join the BA or the BS program. But arts marks are not directly comparable with science marks. For example, suppose that Alice got 81 in arts and is the topper in arts. But she secured 85 in science and is fifteenth with respect to science marks. So the HOD decides to use the class ranks of the students for his decision problem. Let $R_A$ denote a student's arts rank and $R_S$ his/her science rank. For example, Alice has $R_A = 1$ and $R_S = 15$. The student joins the BA program if $R_A < R_S$, the BS program if $R_A > R_S$, and is asked for a choice if $R_A = R_S$. While calculating ranks of students in each subject, students with equal marks are given the same rank. For example, if the top eight science marks are $100, 98, 98, 97, 96, 96, 96, 95$, these students get the ranks $1, 2, 2, 4, 5, 5, 5, 8$, respectively.

Assume that you have at your disposal the functions written for Part 1. Help the HOD to solve his problem by writing a function which outputs the correct decisions for all the students (BA or BS or Student's Choice) in a total of $O(n)$ time. Your function should first print the decisions in the ascending order of the students' roll numbers. Then it should print the number of students for whom the decision is BA, the number of students with BS, and the number with Student's Choice.

**The main() function:**

- Read $n$ (the number of students) from the user. The user then enters the roll numbers, the arts marks, and the science marks for $n$ students, in that order. It is the user's responsibility to enter the data in the ascending order of the roll numbers. Store the entered data in an array of records. After this, the array will be assumed to meet the Department's policy (like read-only, copy-prohibited).

- Process a few queries from the user by invoking the function(s) written for Part 1. Write a loop. In each iteration, the user first enters an option $0, 1, 2$. The option 1 stands for an interval query for arts, 2 a query for science, and 0 for breaking the loop. For an interval query, the boundaries $a$ and $b$ should be additionally supplied by the user.

- Print the HOD's decision for all the students by calling the function of Part 2.

---

**Sample Output**

```
n = 15

+++ Initial list (sorted by roll numbers)
    14SY10022, 71, 63
    15SY10001, 62, 34
    15SY10003, 45, 45
    15SY10004, 11, 24
    15SY10005, 53, 25
    15SY10006, 40, 46
    15SY10008, 77, 85
    15SY30002, 31, 51
    15SY30005, 49, 75
    15SY30006, 70, 85
    15SY71P01, 74, 53
    15SY71P05, 65, 34
    15SY91R01, 33, 71
    15SY91R04, 71, 22
    15SY91R05, 18, 72

+++ Interval search

+++ Arts: a = 53, b = 65
--- List of students:
    15SY71P05, 65, 34
    15SY10001, 62, 34
    15SY10005, 53, 25

+++ Arts: a = 13, b = 14
--- List of students:

+++ Science: a = 69, b = 74
--- List of students:
    15SY91R05, 18, 72
    15SY91R01, 33, 71

+++ Science: a = 26, b = 41
--- List of students:
    15SY10001, 62, 34
    15SY71P05, 65, 34

+++ Rank comparisons
    14SY10022, 71, 63, Arts rank =  3, Science rank =  6, Decision: BA
    15SY10001, 62, 34, Arts rank =  7, Science rank = 11, Decision: BA
    15SY10003, 45, 45, Arts rank = 10, Science rank = 10, Decision: SC
    15SY10004, 11, 24, Arts rank = 15, Science rank = 14, Decision: BS
    15SY10005, 53, 25, Arts rank =  8, Science rank = 13, Decision: BA
    15SY10006, 40, 46, Arts rank = 11, Science rank =  9, Decision: BS
    15SY10008, 77, 85, Arts rank =  1, Science rank =  1, Decision: SC
    15SY30002, 31, 51, Arts rank = 13, Science rank =  8, Decision: BS
    15SY30005, 49, 75, Arts rank =  9, Science rank =  3, Decision: BS
    15SY30006, 70, 85, Arts rank =  5, Science rank =  1, Decision: BS
    15SY71P01, 74, 53, Arts rank =  2, Science rank =  7, Decision: BA
    15SY71P05, 65, 34, Arts rank =  6, Science rank = 11, Decision: BA
    15SY91R01, 33, 71, Arts rank = 12, Science rank =  5, Decision: BS
    15SY91R04, 71, 22, Arts rank =  3, Science rank = 15, Decision: BA
    15SY91R05, 18, 72, Arts rank = 14, Science rank =  4, Decision: BS

    Number of BA = 6
    Number of BS = 7
    Number of SC = 2
```

Submit a single C/C++ file solving all the parts. Do not use global/static variables.