BINUS University

Undergraduate Program *Regular/~~Smart Program/~~ Global Class *)*			**Term : Odd/~~Even~~/~~Short~~ *)**
☐ **Mid Exam** ☒ **Final Exam** ☐ **Short Term Exam** ☐ **Others Exam : _____**			**Academic Year :** **2018 / 2019**
Faculty / Dept.	:	School of Computer Science	Student ID :
Code / Course	:	COMP6047 – Algorithm & Programming (Multipaper II)	
Class	:		N a m e :
Shift	:	Morning	
Day/ Date	:	Friday / 25 January 2019	Signature :
Time	:	13:30 – 15:25 (115 minutes)	
Lecturer	:	Team	
Exam Type	:	~~Open Book~~/Close Book/~~Open e-Book/ Submit Project/Oral Test~~ *)	
Equipment	:	~~Exam Booklet~~/~~Calculator/Dictionary/ Laptop/Tablet/Smartphone/ Drawing Paper A3/Drawing Paper A2~~ *)	
*) *Strikethrough the unnecessary items*			
Please insert the test paper into the exam booklet and submit both papers after the test!!! ***The penalty for CHEATING is DROP OUT!!!***			

Rules:

Students **ARE NOT ALLOWED** to bring:

- Any digital materials (e.g., e-books, softcopy codes, etc.)
- Any digital devices (e.g., flash disks, external hdd, **apple watch or any other smart watches**, etc.)
- Any other electronic devices (e.g., smartphone, handphone, calculator, laptop, etc.)
- Any communication devices should be turned off during this exam.

Fair Examination

A student may be zero-scored for any cheat or attempt of cheat (submitting a solution which is not made by yourself is considered as a very serious cheating) and will be processed through University's policy regarding cheating. A student may also be excused from the exam room for any activity that jeopardized the exam (e.g., **hacking the exam system**, dislodging extension cords, distracting behavior, etc.). All students should uphold the honesty and the spirit of fair examination.

For Indonesian Version of the problems, you can find it after the English Version.

Problem A
Student Record Finder

In this problem, you are given N student records with each student record contains **StudentID**, **Name**, **Age**, **GPA** information. Those N student records are given line-by-line with information separated by **semi-colon** (;) and given in **ascending order** by StudentID. After you read those N student record, there will be M question asked in form of "Is a student with ID X exists?". If the student with ID X exists, you need to print the student information. But if not, you need to print "not exists".

Format Input
The first line of the input consists of a single integer N, denoting the number of student records. After that, N lines followed. Each of those N lines contains the student record information **StudentID**, **Name**, **Age**, **GPA** that are separated by semi-colon (;). After that, there will be a single integer M, denoting the number of question asked. After that, M lines followed. Each of those M lines, contains the asked StudentID.

Format Output
For each M question asked, output a line of student information if the student with the asked studentID exists, or "not exists" if the record not exists. For the GPA, please output 2 digit after comma.

Constraints
$\mathbf{1 \leq N \leq 100.000}$
$\mathbf{1 \leq |Name| \leq 20}$
$\mathbf{0.0 \leq GPA \leq 4.0}$
$\mathbf{10 \leq Age \leq 100}$
$\mathbf{1 \leq M \leq 100.000}$
$\mathbf{N * M \leq 10.000}$ **(80% of cases)**
$\mathbf{N * M \leq 1.000.000}$ **(20% of cases)**

You can assume that the length of the StudentID is always 10, and each digit is a number from 0 - 9.

Sample Input	Sample Output
5 1100789040;Bibi Lili;18;3.95 1200429118;Lili Jojo;22;4.0 1301241234;Jojo Bibi;16;2.75 1310791127;Andrew Wilson;18;3.1 1416709121;Wilson Agata Andi;17;3.9 3 1301241234 1310791128 1416709121	Name: Jojo Bibi ID: 1301241234 Age: 16 GPA: 2.75 not exists Name: Wilson Agata Andi ID: 1416709121 Age: 17 GPA: 3.90

Notes
You'll only get at maximum 80 from the system if your search is **slow** (You will get **TIMELIMIT (80)**).

Even though it's not stated explicitly, you should know by now that excessive space / newline are treated as **WRONG ANSWER**.

Problem B
Student Record Sorting Machine

Given a sequence of N student records, you need to sort the student record (see problem A for the student record details) using **any fast-enough sort algorithms** that you've learnt during this course. After you sort the student record **ascending** by StudentID, you only need to output the student records from index A to B (1-based-index). The Student data are located at a file named **testdata.in**, and you need **to open the file** in **read mode** to do this problem. Please look at Sample Input for clarity.

Format Input
The filename you need to read the data from is **testdata.in.** The first line of the input consists of a single integer N, denoting the number of student records. After that, N lines followed. Each of those N lines contains the student record information **StudentID**, **Name**, **Age**, **GPA** that are separated by semi-colon (;). After that, there are number A and B, where number A is the initial index and number B is the final index.

Format Output
You need to output the student record from index A to B when the student records are sorted by StudentID. Please see sample output for clarity. For the GPA, please output 2 digit after comma.

Constraints
$\mathbf{1 \leq N \leq 1.000}$ **(80% of cases)**
$\mathbf{1 \leq N \leq 100.000}$ **(20% of cases)**
$\mathbf{1 \leq |Name| \leq 20}$
$\mathbf{0.0 \leq GPA \leq 4.0}$
$\mathbf{10 \leq Age \leq 100}$
$\mathbf{1 \leq A \leq B \leq N}$
$\mathbf{B - A \leq 100}$

You can assume that the length of the StudentID is always 10, and each digit is a number from 0 - 9.

Sample Input (testdata.in)	Sample Output
5 1301241234;Jojo Bibi;16;2.75 1200429118;Lili Jojo;22;4.0 1416709121;Wilson Agata Andi;17;3.9 1100789040;Bibi Lili;18;3.95 1310791127;Andrew Wilson;18;3.1 2 4	Name: Lili Jojo ID: 1200429118 Age: 22 GPA: 4.0 Name: Jojo Bibi ID: 1301241234 Age: 16 GPA: 2.75 Name: Andrew Wilson ID: 1310791127 Age: 18 GPA: 3.10

Notes
You'll only get at maximum 80 from the system if your sort function is **slow** to handle the given data (You will get **TIMELIMIT (80)**).
The sample given is the **unsorted version** of Problem A's student record. See Problem A for the **sorted version** by StudentID.

Even though it's not stated explicitly, you should know by now that excessive space / newline are treated as **WRONG ANSWER**.

BINUS University

Undergraduate Program *Regular/~~Smart Program/~~ Global Class *)*			**Term : Odd/~~Even~~/~~Short~~ *)**
☐ **Mid Exam** ☒ **Final Exam** ☐ **Short Term Exam** ☐ **Others Exam : _____**			**Academic Year :** **2018 / 2019**
Faculty / Dept.	:	School of Computer Science	Student ID :
Code / Course	:	COMP6047 – Algorithm & Programming (Multipaper II)	
Class	:		N a m e :
Shift	:	Morning	
Day/ Date	:	Friday / 25 January 2019	Signature :
Time	:	13:30 – 15:25 (115 minutes)	
Lecturer	:	Team	
Exam Type	:	~~Open Book~~/Close Book/~~Open e-Book/ Submit Project/Oral Test~~ *)	
Equipment	:	~~Exam Booklet~~/~~Calculator/Dictionary/ Laptop/Tablet/Smartphone/ Drawing Paper A3/Drawing Paper A2~~ *)	
*) *Strikethrough the unnecessary items*			
Please insert the test paper into the exam booklet and submit both papers after the test!!! ***The penalty for CHEATING is DROP OUT!!!***			

Rules:

Students **ARE NOT ALLOWED** to bring:

- Any digital materials (e.g., e-books, softcopy codes, etc.)
- Any digital devices (e.g., flash disks, external hdd, **apple watch or any other smart watches**, etc.)
- Any other electronic devices (e.g., smartphone, handphone, calculator, laptop, etc.)
- Any communication devices should be turned off during this exam.

Fair Examination

A student may be zero-scored for any cheat or attempt of cheat (submitting a solution which is not made by yourself is considered as a very serious cheating) and will be processed through University's policy regarding cheating. A student may also be excused from the exam room for any activity that jeopardized the exam (e.g., **hacking the exam system**, dislodging extension cords, distracting behavior, etc.). All students should uphold the honesty and the spirit of fair examination.

Problem A
Student Record Finder

Dalam *problem* ini, Anda diberikan N data mahasiswa dimana setiap data berisi informasi **StudentID**, **Name**, **Age**, **GPA**. Semua data mahasiswa diberikan per baris yang dipisahkan oleh **semi-colon** (;) dan data sudah diurutkan secara ***ascending*** berdasarkan **StudentID**. Setelah itu, akan ada pertanyaan M yang ditanyakan dalam bentuk "Apakah mahasiswa dengan ID X ada?". Jika mahasiswa dengan ID X ada, Anda perlu mencetak semua data mahasiswa. Tetapi jika tidak, Anda perlu mencetak "not exists".

Format Input
Baris pertama input terdiri dari sebuah bilangan N, yang menunjukkan jumlah catatan mahasiswa. Setelah itu, terdapat N buah baris yang yang terdiri dari **StudentID**, **Name**, **Age**, **GPA** yang dipisahkan oleh **semi-colon** (;). Kemudian terdapat M buah baris yang menunjukkan jumlah pertanyaan yang diajukan. Masing-Masing baris M berisi **StudentID** yang diminta.

Format Output
Untuk setiap pertanyaan M yang ditanyakan akan memberikan *output* berupa data mahasiswa jika mahasiswa dengan StudentID tersebut ditemukan. Jika StudentID tidak ditemukan, program akan memberikan output "not exists". Untuk IPK yang ditampilkan hanya 2 digit setelah *comma* (,).

Constraints
$\mathbf{1} \leq \boldsymbol{N} \leq \mathbf{100.000}$
$\mathbf{1} \leq |\boldsymbol{Name}| \leq \mathbf{20}$
$\mathbf{0.0} \leq \boldsymbol{GPA} \leq \mathbf{4.0}$
$\mathbf{10} \leq \boldsymbol{Age} \leq \mathbf{100}$
$\mathbf{1} \leq \boldsymbol{M} \leq \mathbf{100.000}$
$\boldsymbol{N} * \boldsymbol{M} \leq \mathbf{10.000}$ **(80% of cases)**
$\boldsymbol{N} * \boldsymbol{M} \leq \mathbf{1.000.000}$ **(20% of cases)**

Anda dapat berasumsi jika panjang dari StudentID akan selalu 10, dan setiap digit merupakan angka dari 0 - 9.

Sample Input	Sample Output
5 1100789040;Bibi Lili;18;3.95 1200429118;Lili Jojo;22;4.0 1301241234;Jojo Bibi;16;2.75 1310791127;Andrew Wilson;18;3.1 1416709121;Wilson Agata Andi;17;3.9 3 1301241234 1310791128 1416709121	Name: Jojo Bibi ID: 1301241234 Age: 16 GPA: 2.75 not exists Name: Wilson Agata Andi ID: 1416709121 Age: 17 GPA: 3.90

Notes
Anda hanya akan mendapatkan maksimum 80 dari sistem jika *searching* yang digunakan **lambat** (Anda akan mendapatkan **TIMELIMIT (80)**).

Meskipun tidak dinyatakan secara eksplisit, Anda harus tahu bahwa *space* / *newline* yang berlebihan diperlakukan sebagai **WRONG ANSWER**.

Problem B
Student Record Sorting Machine

Diberikan urutan N data mahasiswa, Anda perlu mengurutkan data mahasiswa (lihat problem A untuk detail data mahasiswa) menggunakan **algoritma sorting yang cepat** yang telah Anda pelajari selama matakuliah ini. Setelah Anda meng-*sorting* data mahasiswa secara ***ascending*** berdasarkan StudentID, Anda harus menampilkan data mahasiswa dari indeks A ke B (1-based-index). Anda perlu **membaca file** dalam mode ***read*** untuk *problem* ini yang terletak di file **testdata.in**. Silakan lihat *Sample Input* untuk lebih jelas.

Format Input
Nama file yang harus dibaca adalah **testdata.in**. Baris pertama input terdiri dari sebuah bilangan N, yang menunjukkan menunjukkan jumlah catatan mahasiswa. Setelah itu, terdapat N buah baris yang yang terdiri dari **StudentID**, **Name**, **Age**, **GPA** yang dipisahkan oleh *semi-colon* (;). Kemudian terdapat bilangan A dan B, dimana bilangan A sebagai indeks awal dan bilangan B sebagai indeks akhir.

Format Output
Anda perlu menampilkan data mahasiswa dari indeks A ke B setelah data siswa di sorting berdasarkan StudentID. Silakan lihat contoh *sample output* untuk lebih jelas. Untuk GPA yang ditampilkan hanya 2 digit setelah *comma (,)*.

Constraints
$\mathbf{1 \leq N \leq 1.000}$ **(80% of cases)**
$\mathbf{1 \leq N \leq 100.000}$ **(20% of cases)**
$\mathbf{1 \leq |Name| \leq 20}$
$\mathbf{0.0 \leq GPA \leq 4.0}$
$\mathbf{10 \leq Age \leq 100}$
$\mathbf{1 \leq A \leq B \leq N}$
$\mathbf{B - A \leq 100}$
Anda dapat berasumsi jika panjang dari StudentID akan selalu 10, dan setiap digit merupakan angka dari 0 - 9.

Sample Input (testdata.in)	Sample Output
5 1301241234;Jojo Bibi;16;2.75 1200429118;Lili Jojo;22;4.0 1416709121;Wilson Agata Andi;17;3.9 1100789040;Bibi Lili;18;3.95 1310791127;Andrew Wilson;18;3.1 2 4	Name: Lili Jojo ID: 1200429118 Age: 22 GPA: 4,0 Name: Jojo Bibi ID: 1301241234 Age: 16 GPA: 2,75 Name: Andrew Wilson ID: 1310791127 Age: 18 GPA: 3,10

Notes
Anda hanya akan mendapatkan maksimum 80 dari sistem jika *sorting* yang dingunakan **lambat** (Anda akan mendapatkan **TIMELIMIT (80)**). Sampel yang diberikan adalah versi data yang belum di *sorting*. Lihat problem A untuk versi yang sudah diurutkan berdasarkan StudentID.

Meskipun tidak dinyatakan secara eksplisit, Anda harus tahu bahwa *space* / *newline* yang berlebihan diperlakukan sebagai **WRONG ANSWER**.

-- Good Luck & Have Fun --