

Lab Test Example #1

Instructions

- Time to complete the exam: 50 minutes.
- This exam is individual
- This exam is open book
- This exam is closed everything else, including lecture notes, electronic devices, etc
- The only programs you are allowed to have open on the computer during this exam are:
 - The text editor of your choice
 - Terminal window(s)
 - ssh connection to vogon
- Do not use any code you may have access to from earlier in this course or from other courses
- Before starting your work, open a terminal session, and create a directory for the exam. It will be referred to as `labexam01` below.
- The problem description includes all assumptions necessary to answer the problem. Please raise your hand, or approach the instructor if you have any questions.
- Please, collect candy wrappers after you! Additional candy is available, at the instructor's desk: help yourself!
- GOOD LUCK!

Assignment

Your program will take as input information about enrollments at the Computer Science department of a university for up to 20 different years. It will also take as input some queries to which your program will need to provide answers.

Input: enrollment information. Your program shall read input from standard input. The first part of the input will consist of up to 20 lines specifying two *integer* numbers:

1. **Year:** the year for which the enrollment information is provided;
2. **Enrollment:** the number of students enrolled in the department.

The enrollment information will be terminated by a sentinel line

0 0

For example, the following sample input,

```
1994 120
1997 212
1988 103
2000 276
0 0
```

specifies student enrollments in years 1994, 1997, 1988 and 2000. Note that years appear in arbitrary order and need not be consecutive.

Input:queries. The rest of the input to your program will consist of queries. Each query is an integer number representing a year. The queries are terminated by a sentinel line

0

For example, the following *complete* input,

```
1994 120
1997 212
1988 103
2000 276
0 0
1995
2000
1994
0
```

represents information about student enrollments in years 1994, 1997, 1988 and 2000. It contains three questions that the program has to answer: "*What was student enrollemnt in 1995?*", "*What was student enrollment in 2000?*" and "*What was student enrollment in 1994?*".

Program behavior. Your program shall read the first part of the input (the enrollment information) and store it in an array (or collection of arrays). It shall then read, one-by-one, each query and produce an answer to it.

For each query, your program shall search the array (collection) for the year specified in the query. If the year is found, your program shall output the following text:

```
Year YEAR enrollment is ENROLLMENT
```

Where **YEAR** is the year specified in the query, and **ENROLLMENT** is the enrollment number for that year.

If the year *is not found*, your program shall output the following text:

```
Year YEAR enrollment is unknown
```

For example, for the input above, your program shall produce the following output:

```
Year 1995 enrollment is unknown
Year 2000 enrollment is 276
Year 1994 enrollment is 120
```

Program structure. You get to decide how you want to store the enrollment data (parallel arrays, a multidimensional array, etc.). Your program shall include function `int findYear()` which takes as input a year and the array storing the years (you need to provide proper parameter declaration) and outputs the index of array storing the provided year, or -1 if the year is not found in the array.

Error checking. No error checking is necessary, all input files will contain correct inputs.

Program name. Name your program `enrollment.c`

Testing your solution

Compile your program using the following command:

```
> gcc -ansi -Wall -Werror -lm -o enrollment enrollment.c
```

You have access to my full set of test cases for this program, my executable of it and scripts to test my executable and your executable. Copy all the files to your `labexam03` directory with the following command:

```
cp ~dekhtyar/www/101-Winter2011/tr1e11/* .
```

Instructor's executable is called `enrollment-alex`. Do `ls -al` to ensure that it is executable. If not, run `chmod u+x enrollment-alex`. Perform the same check for `enrollment-test.csh` and `enrollment-alex-test.csh` files that are also copied. Test file names are `enroll-test01` through `enroll-test10`.

To test your program, first compile it into an executable named `enrollment`. Then run the following command:

```
> enroll-test.csh
```

The script will run your program on all test cases and produce output. To check it against the instructor's output, open a second terminal session, change to `labexam03` directory and run the command:

```
> enroll-alex-test.csh
```

Compare the two outputs.

Note. You can run the following commands:

```
> enroll-test.csh > my.out
> enroll-alex-test.csh > alex.out
> diff my.out alex.out
```

If the last command does not produce any output, your results exactly match the instructor's results.

Submitting your solution

Once you are satisfied your program meets all the requirements you may turn it in and leave the exam. Run the following command on `vogon` to submit your program:

```
handin dekhtyar labexam03 enroll.c
```