Computer Science

Y. Liow (November 12, 2024)

Contents

Chapter 102

Asymptotics and algorithmic runtime analysis

Open `algorithm-analysis-how-fast-is-an-algorithm.tex` and you'll see

```
\input{exercises/runtime-of-sum-of-squares/main.tex}
```

Goto `exercises/runtime-of-sum-of-squares/` and in that directory you'll see `question.tex` and `answer.tex`. `question.tex` contains the question and `answer.tex` contains the answer. Near the top of `answer.tex`, you'll see

AUTHOR: JOHN DOE. DATETIME: 2024-11-12 09:00:00

For the question you are working on, imitate the above example, replacing John Doe with your name and datetime with the date and time you are done.

To properly generate the table of contents you'll need to run make twice. If you see some errors involving missing python code, let me know.

Once you are done, send me the directory of the exercise, tar'd and gzipped. It should be a directory with `main.tex`, `question.tex` and `answer.tex`. Retain the original directory name. For instance it might be

```
runtime-of-sum-of-squares.tar.gz
```

You'll also need to tell me chapter and course.

102.1 Algorithmic analysis: how fast is an algorithm? debug: algorithm-analysis-how-fast-is-an-algorithm.tex

Some text. Some text. Some text. Some text. Some text. Some text. Some text. Some text. Some text. Some text. Some text.

The following is an example:

Exercise 102.1.1. The following computes the sum of squares from 1^2 to n^2: debug: exercises/runtime-of-sum-of-squares/question.tex

```
s = 0
for i = 1, ..., n:
    term = i * i
    s = s + term
```

Here's the program with goto statements and timing for each statement:

```
                                time
          i = 1                 t1
          s = 0                 t2
LOOP:     if i > n:             t3
              goto ENDLOOP      t4
          term = i * i          t5
          s = s + term          t6
          i = i + 1             t7
          goto LOOP             t8
ENDLOOP:
```

(a) Compute the time taken $T(n)$ as a function of n with constants t_1, ..., t_8.

(b) Simplify the runtime function by giving names A, B, ... to the constants of the function from (a).

(c) Fudge away the constants and write down the simplest $g(n)$ such that the time in (b) is a big-O of your $g(n)$. Your $g(n)$ should be either n or n^2 or n^3 or ...

(d) What is the space complexity of the algorithm?

(Go to solution, page 4004) □

Some text. Some text. Some text. Some text. Some text. Some text. Some text. Some text. Some text. Some text. Some text.

To include your exercise, start by placing it in the `exercises` directory. For example, if your exercise directory is named `tree-6`, you should see the following output when running `ls` inside the `exercises` directory:

```
runtime-of-sum-of-squares  tree-6
```

Next, to display the exercise in this document, from the main directory open:

`algorithm-analysis-how-fast-is-an-algorithm.tex`

then locate the placeholder:

`%???%`

and replace it with:

```
\input{exercises/tree-6/main.tex}
```

Now, go back to the main directory and run the `make` command. This should compile the document with your exercise included.

To add your answer, navigate to the directory of your exercise and enter your solution in the file named `answer.tex`. Return to the main directory, run `make` again, and your solution should appear on the following pages of this document.

Solutions

Solution to Exercise 102.1.1.

debug: exercises/runtime-of-sum-of-squares/answer.tex

AUTHOR: JOHN DOE. DATETIME: 2024-11-12 09:00:00

(a) The following gives the number of times each statement is executed in terms of n:

```
                              time    number of times
         i = 1                t1      1
         s = 0                t2      1
LOOP:    if i > n:            t3      n + 1
             goto ENDLOOP     t4      1
         term = i * i         t5      n
         s = s + term         t6      n
         i = i + 1            t7      n
         goto LOOP            t8      n
ENDLOOP:
```

Therefore the runtime $T(n)$ is

$$T(n) = (t_3 + t_5 + t_6 + t_7 + t_8)n + (t_1 + t_2 + t_3 + t_4)$$

(b)

$$T(n) = An + B$$

where A, B are constants.

(c)

$$T(n) = O(n)$$

(d)

$$\text{SPACE}(n) = O(1)$$

□