

EE231002 Introduction to Programming

Lab01. Minimum Response Time

Due: Sep. 16, 2019

Over 50 years ago, mankind landed on the Moon. That day millions of people watched the live TV broadcasting when Neil Armstrong set his foot on the Moon. During the broadcasting, people observed time lag in the communication between the astronauts and the space center on Earth. This is due to the long distance between Earth and Moon, even light takes more than a second to travel for this distance.

Your assignment is to write a **C** program to calculate the minimum response time given the distance to a planet. It should prompt for an input of distance in Kilometers, and then calculates the minimum response time. The time that the light travels the distance to the planet and back. The speed of light is known to be 299,792,458 meters/second. Example of programming execution is as following.

```
$ ./a.out
Input distance in kilometers: 80000000
The minimum response time is 533.703 seconds.
$ ./a.out
Input distance in kilometers: 380000000
The minimum response time is 2535.09 seconds.
```

Notes.

1. Create a directory **lab01** and use it as the working directory.
2. Name your program source file as **lab01.c**.
3. The first few lines of your program should be comments as the following.

```
// EE231002 Lab01. Minimum Response Time
// Your ID, Name
// Date
```

4. After finishing editing your source file, you can execute the following command to compile the program,

```
$ gcc lab01.c
```

If no compilation errors, the executable file, **a.out**, should be generated, and you can execute it by typing

```
$ ./a.out
```

5. Typical inputs and outputs of the program execution have been shown above. But you should try a few more test cases to make sure your program function correctly.

6. After you finish verifying your program, you can submit your source code by

```
$ ~ee2310/bin/submit lab01 lab01.c
```

If you see a "submitted" message, then you are done. In case you want to check which file and at what time you submitted your labs, you can type in the following command:

```
$ ~ee2310/bin/subrec lab01
```

It will show the last few submission records.

7. The minimum and maximum distances of the Planets in the Solar system to Earth are shown in the table below. You can use this lab to calculate the minimum response time if one day human beings are travelling to those planets.

| Planet | Min distance (kilometers) | Max distance (kilometers) |
|---------|------------------------------|------------------------------|
| Mercury | 80,000,000 | 220,000,000 |
| Venus | 42,000,000 | 258,000,000 |
| Mars | 80,000,000 | 380,000,000 |
| Jupiter | 628,000,000 | 928,000,000 |
| Saturn | 1,250,000,000 | 1,550,000,000 |
| Uranus | 2,850,000,000 | 3,150,000,000 |
| Neptune | 4,350,000,000 | 4,650,000,000 |
| Pluto | 4,250,000,000 | 7,550,000,000 |

Coding Guidelines

1. Use **comments** to explain your codes.
 - 1.1. Header comments are needed at the beginning of a file.
 - 1.2. Global variables and function declarations need to have comments.
 - 1.3. Key operations must be clearly documented.
 - 1.4. Spelling must be correct.
 - 1.5. Comments should also be properly indented and with space characters inserted.
2. Use **indentation** to group statements at the same block level.
 - 2.1. Use <tab> for indentation.
3. Use **blank lines** to separate
 - 3.1. directives and functions,
 - 3.2. declarations and statements.
 - 3.3. All declarations must precede statements in a function.
4. Use **space character** to separate tokens.
 - 4.1. The same way as in English sentences.
5. **Variable name** should be descriptive.
 - 5.1. *i, j, k* for integral local temporary variables,
 - 5.2. *x, y, z* for floating point local temporary variables,
 - 5.3. *p, q, r* for local temporary pointers.
 - 5.4. All-capital tokens for symbolic constants.
6. Each line of source code **should not** have more than **80 characters**.