VIETNAM NATIONAL UNIVERSITY - HO CHI MINH CITY
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

PROGRAMMING FUNDAMENTALS - CO1027

ASSIGNMENT 1

SHERLOCK HOLMES
THE AWAKENED - Part 1

HO CHI MINH CITY, NOVEMBER 2023

ASSIGNMENT SPECIFICATIONS

Version 1.1

1 Outcomes

After finishing this assignment, the student is revised and can proficiently use:

- Conditional statements
- Loop statements
- Array and 2-dimensional array
- String processing
- Function and function call
- File read/write operations

2 Introduction

The assignment is adapted from the video game Sherlock Holmes: The Awakened, developed by Frogwares

6 September 1894, a bored Sherlock Holmes stares out his window. The detective is going through another frustrating dry spell where no case has presented itself worthy of his talents. He's about to get his wish.

3 Tasks

Students are asked to implement a program in C++ to simulate the solving process of Sherlock Holmes: The Awakened, through the tasks described below. Each task requires writing a corresponding function, and the parameters for this function will be provided in the task description.

3.1 The First Case: The Dock Abnormalities

Word soon reaches him of the disappearance of a young Maori manservant who works for the wealthy Captain Stenwick. Following the clues takes them to the docks near the River Thames, where Holmes and John Watson soon learn that many similar 'disappearances' have occurred.

When they arrived at the dock, Holmes examined its documents, which are records of ships arriving and departing the docks. However, the explosion of the Red Ship caused the records to be burned and disorganized. Holmes and Watson sifted through the burned fragments and discovered irregularly forged dock documents.

3.1.1 The Invalid Time

Students are asked to write a function to find abnormal times in dock documents. The description of the function is as follows:

- Function name: **invalidTime**.
- Input parameters:
  - **schedule**: a string containing filename, this file contains information of the dock schedule.
- Structure of the file **schedule**:
  - Line 1: a formatted string **Number of ships: <n>**, in which **<n>** has 3 numeric characters. **<n>** will represent an integer in the range $[1, 999]$. For example, $n1$ = "001" represents the integer 1; $n1$ = "010" represents the integer 10; $n1$ = "123" represents the integer 123.
  - From Line 2 onwards, each line will be formatted according to the following syntax:

    ship_name, DD/MM hh:mm - DD/MM hh:mm

    In which:
    - ship_name: The name of the ship, represented by uppercase and lowercase letters, and digits (no spaces).
    - DD: Represents the day of the month, represented by 2 digits, with a range of values from 01 to 31. The number of days in the month depends on the current month.
    - MM: Represents the month, represented by 2 digits, with a range of values from 01 to 12.
    - hh: Represents the hour of the day, represented by 2 digits, with a range of values from 00 to 23.
    - mm: Represents the minutes, represented by 2 digits, with a range of values from 00 to 59.
    - A combination of DD/MM hh:mm is considered a time. The first time is the time of arrival, and the second time is the time of departure.

Due to the charred pieces of paper, some digits in the documents are damaged and not legible. These illegible digits are denoted by the character '?'.

- Function requirements: The function reads data from the file **schedule**. If <n> is an invalid string, the function does nothing and returns the value -1. If the string <n> is valid, we count the number of ships that have all valid times for arrival and departure. If there are '?' characters in the schedule, and a substitution can be made for all '?' characters with digits in such a way that the arrival and departure times become valid, then that ship is considered valid. In this function, students are not required to check whether the time interval between arrival and departure is valid; this will be considered in the following section. The function returns the number of valid ships in the schedule file. If there are no valid ships, the function returns -1. The number of days in the month is calculated according to the year 1864.

Example 1: Examples of <**n**> being an invalid string:

- **Number of ships: 11x**: invalid because the last character is 'x', not a digit.
- **Number of ships: 000**: invalid because when parsed to an integer will be zero, not in the range $[1, 999]$.

The function returns the value -1 in those two cases.

Example 2: The file **schedule** has the following content:

1	Number of ships: 004
2	NO22, 12/02 14:00 - 12/02 18:00
3	NVD34, 14/0? 16:1? - ??/03 ?2:00
4	13Se, 15/13 16:00 - 12/01 14:61
5	ko772, 4?/1? ??:?? - 12/05 3?:25

Accordingly:

- Ship on line 2: All times are valid. Therefore, this ship is valid.
- Ship on line 3: We can substitute the '?' characters with 14/03 16:15 - 14/03 12:00. Therefore, this ship is valid.
- Ship on line 4: Invalid because there is no month 13, and the number of minutes exceeds the limit of 59.
- Ship on line 5: Invalid because there is no way to substitute the '?' characters to make the arrival and departure times valid.

The function returns value: 2.

3.1.2 The Longest Duration

After eliminating the invalid ships, Holmes continues to observe suspicious ships by determining which ship has stayed in the dock for the longest duration.

Students are asked to write a function to find the longest duration in dock documents. The description of the function is as follows:

- Function name: **longestDuration**.
- Input parameters:
  - **schedule**: a string containing filename, this file contains information of the schedule.
- Structure of the file **schedule**: Similar to the function **invalidTime**.
- Function requirements: The function reads data from the file **schedule**. If <**n**> is an invalid string, the function does nothing and returns an empty string (""). If the string <**n**> is valid, we find the ship stays at the dock with the longest duration among all ships. Specifically:
  - If the arrival and departure times of the ship are valid, the ship's duration at the dock is equal to the difference between the departure and arrival times.
  - If there are '?' characters in the schedule, we must find a way to substitute these '?' characters to maximize the duration the ship spends at the dock. In other words, we need to find a way to maximize the difference between the two times.
  - If the ship's arrival time at the dock is greater than its departure time, the ship is invalid. If no valid ships exist, the function returns an empty string ("").
  - If there is a ship with the longest duration at the dock, the function returns the name of that ship. If there is more than one ship with the longest duration at the dock, the function returns the name of the last ship in the list with the longest duration.
  - The number of days in the month is calculated according to the year 1864.

Example 3: The file **schedule** has the following content:

Line	Contents	Longest duration (in minutes)
1	Number of ships: 007	
2	NO22, 12/02 14:00 - 12/02 18:00	240
3	NVD34, 14/0? 16:1? - ??/03 ?2:00	111230
4	13Se, 15/13 16:00 - 12/01 14:61	Invalid
5	ko772, 4?/1? ??:?? - 12/05 3?:25	Invalid
6	tyt8, 11/1? ??:?? - 12/04 1?:25	Invalid
7	bor08, ??/?? ??:?? - 12/04 23:25	148285
8	gate12, ??/?? ??:?? - 12/04 23:25	148285

Accordingly:

- Ship on line 3: We can substitute the '?' characters to achieve the longest duration at the dock as follows: 14/01 16:10 - 31/03 22:00. The time at the dock is 111230 minutes.
- Ship on line 6: **Invalid** because there is no way to substitute the '?' characters to make the departure time greater than the arrival time.
- Ships on lines 7 and 8: We can substitute the '?' characters to achieve the longest duration as follows: 01/01 00:00 - 12/04 23:25. The duration of staying is 148285 minutes.

The function returns value: gate12.

3.2 The Second Case: The Abandoned Temple

Holmes and Watson's investigation at the docks leads them to a network of underground tunnels, which they believe are being used to transport the kidnapped Maori manservant and other victims to the abandoned subterranean temple.

The tunnels are dark and labyrinthine, and Holmes and Watson must use their keen senses of observation and deduction to navigate them safely. Along the way, they encounter evidence of a gruesome ritual, including bloodstains, sacrificial symbols, and discarded clothing.

3.2.1 Investigate the temple

Students are tasked with writing the following function to describe the process of Holmes and Watson investigating the temple. The function information is as follows:

- Function name: **investigateTemple**.
- Input parameters:
  - `map`: a string containing filename, this file contains the map of the temple.
  - `moveStyle`: The string contains the moving behavior of Holmes and Watson. Each character is one of the following four letters:
    - 'U': go up
    - 'D': go down
    - 'L': turn left
    - 'R': turn right

    Notably, Holmes and Watson will move with a repeating movement behavior. In other words, if Holmes and Watson have a moveStyle of "LLUU," then they will first turn left twice (LL), then move up twice (UU), and then repeat this sequence indefinitely. Therefore, their actual movement sequence is "LLUULLUULLUULLUULLUU...".
  - `stamina`: An integer representing the physical condition of Holmes and Watson.
  - `outTime`: An integer variable passed by reference, used to return the time it took Holmes and Watson to reach the altar.
  - `outPath`: A string variable passed by reference, used to return the path that Holmes and Watson have walked through.
- Function requirements:
  1. Read data from the file named `map` and initialize appropriate values.
  2. Based on the variable `moves`, the student represents Holmes and Watson's movements sequentially through the cells on the map. While moving, if Holmes and Watson reach a row/column that is the boundary of the map and the next step in `moves` leads them outside the map, they will be transported to the opposite end of the map. For example, if Holmes and Watson are at position (0,0) and move upwards, their next position will be (9, 0).
  3. With each step, the stamina decreases by 1 unit. If Holmes and Watson run out of stamina, they cannot continue moving.
  4. If Holmes and Watson move into the same cell more than 3 times, they will be lost and cannot continue moving.
  5. Holmes and Watson will stop moving if they find the altar, run out of stamina, or encounter certain events during the investigation.
- Return results:
  - The function returns `true` if Holmes and Watson reach the altar. Otherwise, it

returns `false`.

- `outTime`: The integer represents the time it takes Holmes and Watson to reach the altar. Each step will be counted as a unit of time.
- `outPath`: The string represents the path that Sherlock and Watson have walked through. Each point on the path is separated by a character ';'. A path is represented as:

$$p_{i1}, p_{j1}; p_{i2}, p_{j2}; ...; p_{ik}, p_{jk}; ...; p_{in}, p_{jn}$$

where p_{ik}, p_{jk} is the index of a k^{th} point by row and by column, and n is the number of moves that Sherlock has walked, respectively.

3.2.2 The Map of The Temple

The map contains a fixed-size 10x10 two-dimensional array. Each element of the array is a character. This 2D array represents the map of the investigated temple.

The table below represents the objects/events in the temple map:

Character	Object/Event Name	Quantity	Meaning
S	Holmes and Watson	1	Starting position of Holmes and Watson
A	Altar	1	Location of the altar
0	Pathway	Unlimited	Accessible pathways within the temple
X	Obstacle	Unlimited	Impassable obstacles
R	Mirror	Max 1	Reverses movement behaviors
T	Trap Door	Max 2	Automatically moves to the other trap door
H	Time Machine	Max 1	Transports Holmes and Watson to a point in history
P	Potion	Max 1	Reduces 1 stamina for each 2 steps (instead of 1)
C	Cryptic Symbols	Max 1	Rotates the map 180 degrees
L	Light	Max 1	Illuminates obstacles
M	Monster	Max 1	Consumes the investigator's mind, investigation fails

All characters are demonstrated in detail as follows:

- Character 'S': Starting position of Holmes and Watson. Do not save this position to outPath at the beginning.
- Character 'A': Altar position. If you move to this point, the investigation will stop, and it will be considered successful.
- Character '0': This is the path inside the temple.
- Character 'X': This is an obstacle in the temple. Cannot move to these positions. When entering this cell, do not save it to outPath because it is impassable.
- Character 'R': If you encounter a mirror (character R), the mirror will reverse the movement behavior of Holmes and Watson. If the movement behavior of Holmes and Watson

has 5 characters and they are moving on the 3rd step out of 5 characters, then after reversing the string of movement behavior, Holmes and Watson will continue moving on the 4th step of the reversed string.

- Character 'T': It is known that there are exactly 2 trap doors in the temple. If Holmes and Watson move into a trap door, they will be transferred to the other trap door. Only save the position of the trap door entered, do not save the position of the other trap door to outPath or the cells passed.
- Character 'H': Holmes and Watson are time-shifted back to where they have been in the past. It is known that this is the place where, at that time, Holmes and Watson had consumed 1/3 of their initial stamina. If they have not consumed 1/3 of their initial stamina, Holmes and Watson will return to the starting position. After returning to this point, Holmes and Watson will continue moving to the next step in their current movement behavior. Note that the cells they have passed will be reset to match the time in the past. That is, if a certain cell has been passed in the past, when Holmes and Watson are shifted back to the past, that cell will be considered not passed. In terms of reality:
  - Holmes returns to a position in the past.
  - The map is reverted to the past.
  - The list of cells passed is updated to correspond to the past.
  - After performing the event's effects, we only reduce stamina (if any). Then, if stamina is lost by 1/3, this is the past time we need to return to when encountering the 'H' Time Machine.
  - Interaction with the 'P' Medicine Bottle, 'R' Mirror: When returning to the past, we still maintain the effects of the Medicine Bottle and Mirror.
  - outPath, stamina, outTime will remain unchanged when returning to the past.
- Character 'P': If Holmes and Watson find a medicine bottle in the temple, they will be granted mystical power. Specifically, from the moment they drink the medicine, they will reduce 1 stamina when taking 2 steps (instead of 1 step before). Note that the stamina of Holmes and Watson will only decrease by 1 if they take exactly 2 steps. This effect does not accumulate.
- Character 'C': Holmes and Watson have found a series of mysterious symbols and signs carved on the wall during their investigation. When they touch these symbols, the map will immediately rotate 180 degrees. No need to update outPath to match the rotated map. However, the list of cells passed must be updated.
- Character 'L': While investigating the temple, Holmes and Watson find a hidden button inside a cave. When pressed, obstacles will automatically light up by the torches attached

to them within a range of 1 square (including diagonal cells), the illuminated range will not extend beyond the other side of the map. At this point, if the altar is lit by a torch, Holmes and Watson have found the altar, and the investigation ends successfully.

- Character 'M': Unfortunately, if Holmes and Watson step on the temple's trap switch, they will summon the Cthulhu monster and be mentally swallowed by it, and the investigation will end.

3.3 The Third Case: The Locked Crate

After investigating the victim on the altar, Holmes and Watson also find drugs, which leads Holmes to deduce that it is being used as a kind of soporific to subdue victims so that they can be abducted and trafficked out of the country. Besides those drugs, Holmes finds a crate locked by a 3-digit combination lock. Next to the crate is a letter from Dr. Gygax, with a note that includes a triangle, a sequence of numbers, and a number written in blood. Holmes guesses that there must be a connection between these three pieces of evidence to open the lock.

Students are asked to write the following function to help Sherlock open the lock:

- Function name: **solveCrate**.
- Input parameters:
  - `arr`: An array of positive integers with values from 1 to 999.
  - `size`: The size of the array, with values from 1 to 50.
  - `k`: An integer with a value less than `size`.
- Function requirements: Build a triangle of arrays of numbers with the following rules:
  - The first row of the triangle will be the original array.
  - Starting from the second row onwards, the number of elements in the array will decrease by 1 compared to the previous row. At the same time, each element in this row will be equal to the sum of the two adjacent elements in the previous row.

  The code of the crate is the sum of the kth row; if the number is greater than 3 digits, take the last 3 digits.
- Return results: The function returns the last 3 digits of the sum of the k-th row.

Example 4: Given that the **arr** = {1, 2, 3, 4, 5}, n = 5. Then, the triangle of the array would be:
1st row: {1, 2, 3, 4, 5}

2nd row: {3, 5, 7, 9}
3rd row: {8, 12, 16}
4th row: {20, 28}
5th row: {48}

If k is equal to 2, the function returns the value 24.

3.4 Ending

After opening the crate and finding a note labeled "Black Edelweiss Institute", Holmes recalls that the mountain flower Edelweiss (Leontopodium alpinum) is a national symbol of Switzerland.

Sherlock soon realizes that there is much more to this case than even he could've imagined when he initially began. His search for answers will take him not only to 'The Continent' but across the Atlantic to America itself, unraveling a dark plot along the way.

4 Submission

Students submit a file: **awakened.h** in the site "Ky thuat lap trinh (CO1027)_HK231_ALL".

Deadlines for submission are announced at the submission site above. By the deadline for submission, the link will be locked automatically, so students will not be able to submit them late. To avoid possible risks at the time of submission, students MUST submit their papers at least **one hour** before the deadline.

5 Harmony for Assignment

The final exam of the course will have some Harmony questions related to the content of the assignment. Assume the score a student achieves for the assignment is **a** (on a 10-point scale), and the total score for the Harmony questions is **b** (on a 5-point scale). Let **x** be the score of the assignment after Harmony, which is also the final score of the assignment for the student. The final exam questions will be harmonized with 50% of the assignment score using the following formula:

- If $a = 0$ or $b = 0$ then $x = 0$

- If a and b both different from 0 then

$$x = \frac{a}{2} + HARM(\frac{a}{2}, b)$$

Where:

$$HARM(x, y) = \frac{2xy}{x + y}$$

Students must complete the assignment on their own. If a student cheats on the assignment, they will be unable to answer the Harmony questions and will receive a score of 0 for the assignment.

Students **must** pay attention to answering the Harmony questions in the final exam. Those who do not answer them will be considered to have scored 0 for the assignment and will fail the course. **No explanations will be accepted and there are no exceptions.**

6 Handling fraud

Assignment must be done BY YOURSELF. Students will be considered fraudulent if:

- There is an unusual similarity between the source code of the submissions. In this case, ALL submissions are considered fraudulent. Therefore, students must protect the source code of their assignments.
- Students do not understand the source code written by themselves, except for the parts of the code provided in the initialization program. Students can consult from any source, but make sure they understand the meaning of all the lines they write. In the case of not understanding the source code of the place they refer, students are especially warned NOT to use this source code; instead use what has been learned to write programs.
- Mistakenly submit another student's assignment on your personal account.

In the case of cheating, students will get a 0 for the entire subject (not just the assignment).

DO NOT ACCEPT ANY INTERPRETATION AND NO EXCEPTION!

After each major assignment has been submitted, a number of students will be called for random interviews to prove that the assignment has been done by themselves.

7 Change from previous version

- **Section 3.2.1:** Update the next position when leaving the map (0,9) -> (9,0).

- **Section 3.2.2:** Update the quantity of entities/events in the temple map.
- **Section 3.2.2:** Clarify the effects of characters in the map.
- **Section 3.3:** Adjust the range of values for the input parameter `k`.

——————————END——————————