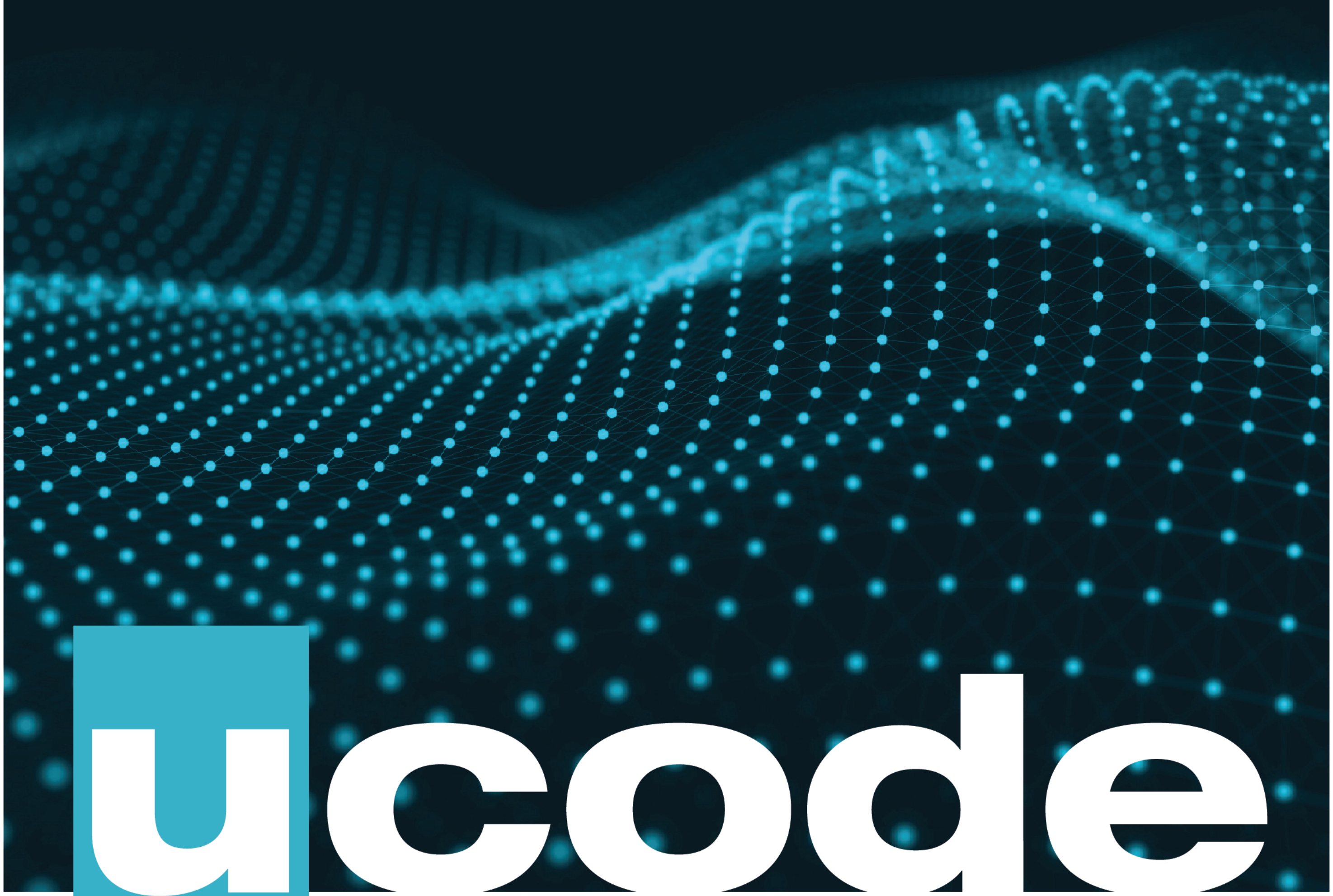
Marathon C

Race 04

June 20, 2019



ucode

Contents

Engage 2
Investigate 3
Act 5
Publishing 8

ucode

Engage

DESCRIPTION

{some cool and motivational words}
{pls change me}

BIG IDEA

Learn to accept difficult challenges.

ESSENTIAL QUESTION

How to motivate yourself and others?

CHALLENGE

Development of the algorithm in the team.

Investigate

GUIDING QUESTIONS

We invite you to find answers to the following questions. This will help you realize what knowledge you will get from this challenge and how to move forward.

Ask your teammates and discuss the following questions together. You can find the answers in the Internet and share it with student around you.

We encourage you to ask as many questions about programming in teams as possible. Note down your discussion.

- How to organize work on the challenge productively?
- What motivated me to grow up in coding?
- How to distribute tasks in a small team?
- What workflow best for the small developers' team?
- What main skill needed for team challenges?

GUIDING ACTIVITIES

These are only a set of example activities and resources. Do not forget that you have a limited time to overcome the challenge. Use it wisely. Remember that all team member must be involved into the process.

1. Read about motivation. Watch a few videos on YouTube on this topic.
2. Find your own personal reasons to learn and grow in programming?
3. Involve teammate in the research process.
4. Read about branching and merging in `git`. This features improve your team time of developming.
5. Get yourself into your teammate's place, which features you would like to see in yourself. How do you need to start talking about a challenge?
6. Try to create positive feelings in the team. Do not be shy, but do not be subjected. Your partner may have skills different from yours. Mutual assistance is always recalled as a result.

ANALYSIS

You need to analyze all the collected information before you start.

- Race has to be carried out by an entire team.
- Each team member must understand the challenge and realization, and to be able to reproduce it individually.

- It is your responsibility to assemble the whole team. Phone calls, SMS, messengers are good ways to stay in touch.
- Be attentive to all statements of the story. Examine the given examples carefully. They may contain details that are not mentioned in the task.
- Perform only those tasks that are given in the story.
- You should submit only the specified files in the required directory and nothing else. In case you are allowed to submit any files to complete the task you should submit only useful files. Garbage shall not pass.
- You should compile C files with clang compiler and use these flags: `-std=c11 -Wall -Wextra -Werror -Wpedantic`.
- You should use only functions which allowed in a certain task.
- Usage of forbidden functions is considered as cheat and your challenge will be failed.
- You must complete tasks according to the rules specified in `the Auditor`.
- Your exercises will be checked and graded by students. The same as you. `Peer-to-Peer (P2P) learning`.
- Also, your exercises will pass automatic evaluation which is called `Oracle`.
- Got a question or you do not understand something? Ask the students or just Google that.
- Use your brain and follow the white rabbit to prove that you are the Chosen one!!!

Act

SOLUTION DEVELOPMENT

Let's get started! And may the odds be ever in your Favor!

1. Meet the team. Talk about the early sprints and how your abilities have grow over these days.
2. Read the story, taking everyone's ideas on board, not just your own.
3. Try to choose the best solution. Distribute the task.
4. Clone your git repository, what is issued on the challenge page in the LMS.
5. Start to develop solution. Offer improvement. Test your code.
6. You must distribute tasks within a team. If it's possible.
7. Explore new things for you. Talk, talk and talk again. Have fun:)

NAME

Way home

DIRECTORY

`./`

SUBMIT

`Makefile, inc/*.[h], src/*.[c]`

ALLOWED FUNCTIONS

`read, write, malloc, free, exit, open, close`

LEGEND

This is my ship...the Nebuchadnezzar, it's a hovercraft. Help me to find the path to the Zion.

DESCRIPTION

Create a program that finds the shortest path in the maze between the entry and exit points using only orthogonal directions.
A maze is specified as a comma-separated sequence of obstacles and empty spaces. The obstacle defined by `#` character and empty space by `.` character.

The file name, entry point (x1, y1) and exit point (x2, y2) coordinates are specified as command-line arguments.
The binary must be called `race04`.

- Calculate the shortest path from the entry point to the most distant point (x3, y3) of the maze. Print the number of movements from (x1, y1) to (x3, y3) on the standard output in `dist=XX` format.
- Calculate the shortest path from the entry point to the exit point. Print the number of movements from (x1, y1) to (x2, y2) on the standard output in `exit=XX` format.
- Save the maze and the path from the entry point to the exit point in the file `path.txt`. Note, that `path.txt` will be assessed only by peers. Obstacles - `#`, empty space - `.`, path - `*`, most distant point - `X` if it is a part of the path or `D` in other cases.

Error handling. Print errors on the standard error `stderr`:

- Print `usage: ./race04 [file_name] [x1] [y1] [x2] [y2]` if a number of command-line arguments are less or more than required.
- Print `map does not exist` in case of missing or empty file.
- Print `map error` if there are forbidden characters or map is not rectangular.
- Print `points are out of map range` if points are out of map coordinates.
- Print `entry point cannot be an obstacle` if the entry point is also an obstacle.
- Print `exit point cannot be an obstacle` if the exit point is also an obstacle.
- Print `route not found` if there is no pass between points.
- Print `error` in case of any other errors.

If there are more than 1 error, output only 1 error message according to priority listed above.

Please see detailed example in CONSOLE OUTPUT section.

CONSOLE OUTPUT

```
>./race04 | cat -e
usage: ./race04 [file_name] [x1] [y1] [x2] [y2]
>cat -e maps/maze.csv
#,#,#,#,#,#,.,#,.,.,.$
#,.,#,.,.,#,.,#,#,#,#$
#,.,#,.,.,.,.,.,.,#,#$
#,.,#,.,.,#,#,#,#,#,#$
#,.,.,#,.,.,.,.,.,.,#$
#,.,.,.,#,.,#,#,.,.,#$
#,.,.,.,.,#,.,.,.,.,#$
#,.,.,.,.,#,.,.,.,.,#$
#,.,.,.,.,.,.,.,.,.,#$
#,#,#,#,#,#,#,#,#,.,#$
>./race04 maps/maze.csv 1 1 3 3 | cat -e
dist=28$
exit=24$
>cat -e path.txt
######D#...$
#*#..#.####$
#*#.....D##$
#*#**######$
#*.#*****.#$
#*..#.##*.#$
#*...#***.#$
#*...#*...#$
#******...#$
#########.#$
>./race04 maps/maze.csv 1 1 8 2 | cat -e
dist=28$
exit=28$
>cat -e path.txt
######D#...$
#*#..#.####$
#*#.****X##$
#*#.*######$
#*.#*****.#$
#*..#.##*.#$
#*...#***.#$
#*...#*...#$
#******...#$
#########.#$
>./race04 maps/maze.csv 0 0 3 3 | cat -e
entry point cannot be an obstacle
>./race04 maps/maze.csv 1 1 0 3 | cat -e
exit point cannot be an obstacle
>./race04 maps/maze.csv 1 1 8 0 | cat -e
route not found
>./race04 maps/maze.csv 1 1 8 22 | cat -e
points are out of map range
>
```

Publishing

PUBLISHING

The final important and integral stage of your work is its publishing. This allows you to share your challenges, solutions, and reflections with a local and global audience.

During this stage, you will find how to get a global assessment. You will get representative feedback. As a result, you get the maximum experience from the work you have done.

What you can create to disseminate information

- Text post, summary from reflection.
- Charts, infographics or any other ways to visualize your information.
- Video of your work, reflection video.
- Audio podcast. You can record a story with your experience.
- Photos from Ucode with small post.

Example techniques

- Canva - a good way to visualize your data.
- QuickTime - easy way to record your screen, capture video, or record audio.

Example ways to share your experience

- Facebook - create a post that will inspire your friends.
- YouTube - upload a video.
- GitHub - share your solution.
- Telegraph - create a post. This is a good way to share information in a Telegram.
- Instagram - share a photos and stories from Ucode. Don't forget to tag us :)

Share what you learned with your local community and the world. Use #Ucode and #CBLWorld on social media.