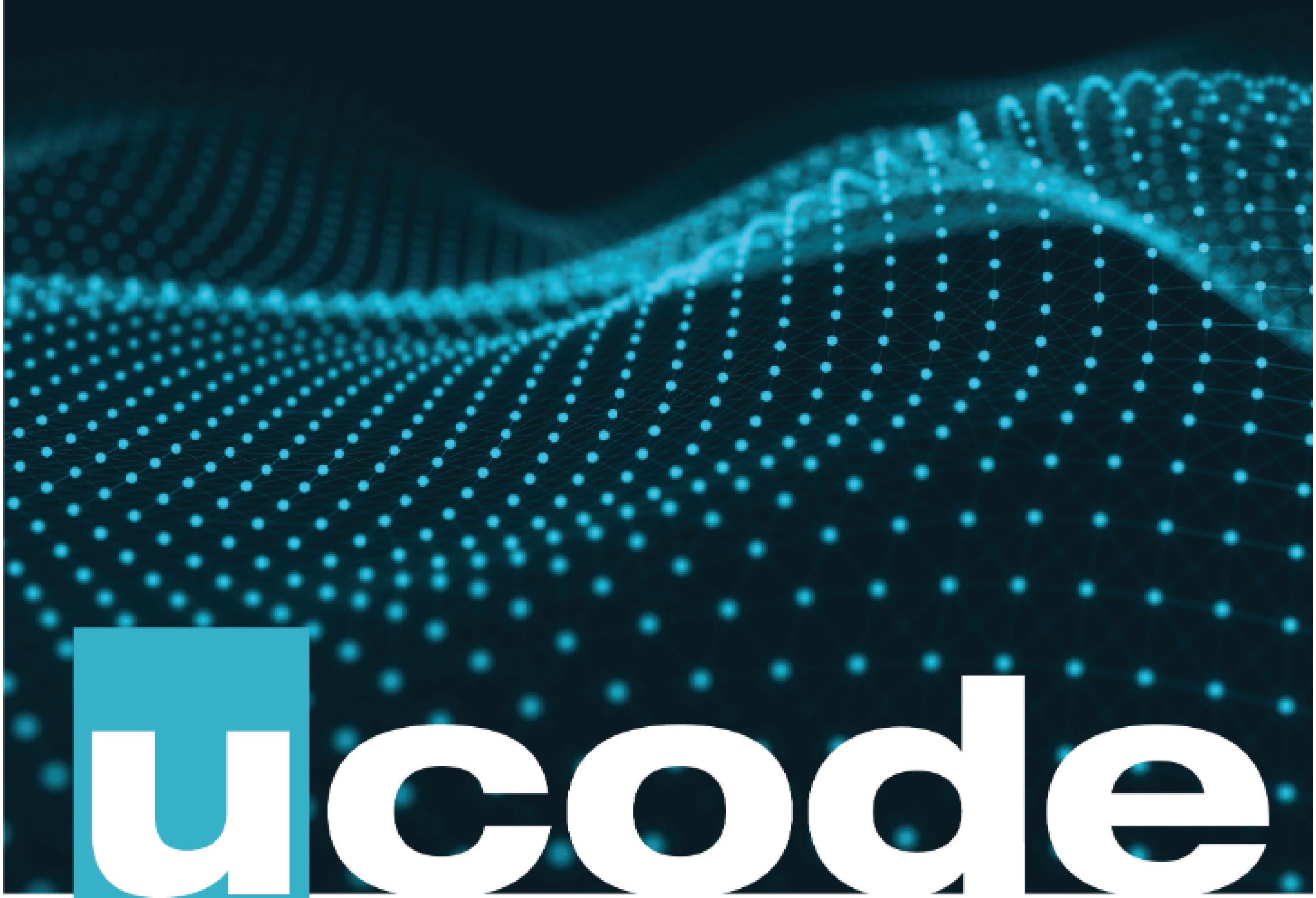
Race 03

Marathon C

April 15, 2020



ucode

Contents

ucode

Engage

DESCRIPTION

New day, new race...
How is it going?

Do you like card games? Have you ever played poker?

Before you dive into building the solution, we invite you to investigate data analysis. Data analysis is a process for obtaining raw data and converting it into information that is useful for decision-making. Data are collected and analyzed to answer questions, test hypotheses, and prove or disprove theories.

Efficient data analysis is an iterative process. It consists of several essential parts. The analysis is preceded by the following steps: definition of analysis requirements, data collection according to the requirements, data processing and organization, and, finally, cleaning redundant values. Only after these steps data can be analysed. Analysts may apply a variety of techniques. Explore which approaches suit this particular challenge best. Choose appropriate algorithms. The result of your data analysis will be a computer app that takes data input and generates output with conclusions.

Would you like to apply in practice the above-mentioned technique?
A lot of data science topics describe complex concepts using simple examples, such as rolling dice, or playing cards.
So, how about making a data analysis of poker hand combinations?
Let's play!

BIG IDEA

Data analysis.

ESSENTIAL QUESTION

How to process data effectively?

CHALLENGE

Analyze poker hand combinations.

Investigate

GUIDING QUESTIONS

We invite you to find answers to the following questions. By researching and answering them, you will gain the knowledge necessary to complete the challenge. To find answers, ask the students in the Slack and search the internet. We encourage you to ask as many questions as possible. Note down your findings and discuss them with your peers.

- How do you prepare for an upcoming Race?
- What card games do you know?
- What are the rules of poker?
- What are poker hand combinations?
- What is the best workflow for a small developer team?
- What is the main skill needed for team challenges?
- How to compute poker hands probability?
- How to find an efficient algorithm?

GUIDING ACTIVITIES

Complete the following activities. Don't forget that you have a limited time to overcome the challenge. Use it wisely. Distribute tasks correctly.

- Meet with your team in the Zoom. Discuss how you will organize teamwork. Create a channel for a team communication.
- Read about motivation. Watch a few videos on YouTube about this topic.
- Write down on a piece of paper your own personal reasons to learn and grow in programming.
- Involve all teammates in the research process.
- Read about branching and merging in `git`. Share your experience. These features improve your development efficiency.
- If you have ever played poker, refresh the rules and all the hand combinations in memory. If not, research the rules and combinations thoroughly.
- Read the story, taking everyone's ideas on board.
- Clone your git repository that is issued on the challenge page in the LMS.
- Distribute tasks among all team members.
- Start to develop the solution. Offer improvements. Test your code.
- Communicate with students and share information.

ANALYSIS

Analyze your findings. What conclusions have you made after completing guiding questions and activities? In addition to your thoughts and conclusions, here are some more analysis results.

- Challenge has to be carried out by the entire team.
- Each team member must understand the challenge and realization, and 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.
- Analyze all information you have collected during the preparation stages.
- Perform only those tasks that are given in this document.
- Submit your files using the layout described in the story. Only useful files allowed, garbage shall not pass!
- Compile C-files with clang compiler and use these flags: `clang -std=c11 -Wall -Wextra -Werror -Wpedantic`.
- Your program must manage memory allocations correctly. Memory which is no longer needed must be released otherwise the task is considered as incomplete.
- Pay attention to what is allowed in a certain task. Use of forbidden stuff is considered a cheat and your tasks will be failed.
- Complete tasks according to the rules specified in the `Auditor`.
- The solution will be checked and graded by students like you. Peer-to-Peer learning.
- Also, the challenge will pass automatic evaluation which is called `Oracle`.
- If you have any questions or don't understand something, ask other students or just Google it.
- Use your brain and follow the white rabbit to prove that you are the Chosen one!

Act

NAME

Poker face

DIRECTORY

`./`

SUBMIT

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

ALLOWED FUNCTIONS

`free, malloc, write, exit`

BINARY

`race03`

LEGEND

There is no Spoon.

DESCRIPTION

Create a program that recognizes poker card combinations of 5-card poker hands. There is a standard deck of 52-cards.
Individual cards are ranked: `2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K, A`.
There are 4 card suits: `H, C, S, D`.

CONSOLE OUTPUT for valid combinations:

- Royal flush
- Straight flush
- Four of a kind
- Full house
- Flush
- Straight
- Three of a kind
- Two pair
- One pair
- No pair

CONSOLE OUTPUT for invalid command-line arguments:

- usage: ./race03 [card1] [card2] [card3] [card4] [card5]
- Duplicate cards: <value>
- Invalid card: <value>

```
Straight flush$
>
```

SEE ALSO

Poker probability
Standard 52-card deck