

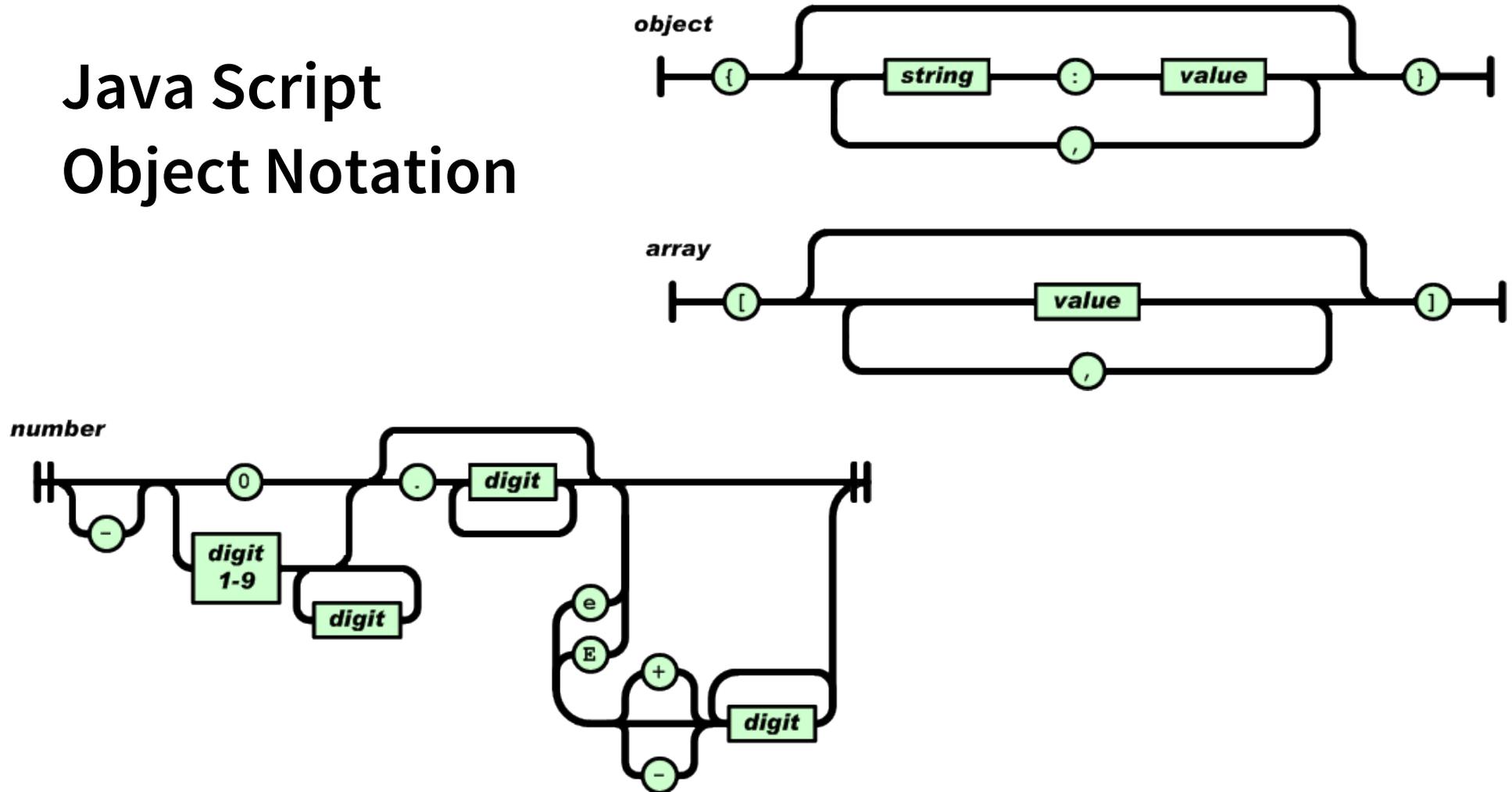
JSON parser

<https://github.com/hhhj123/PA193-parser/tree/master>

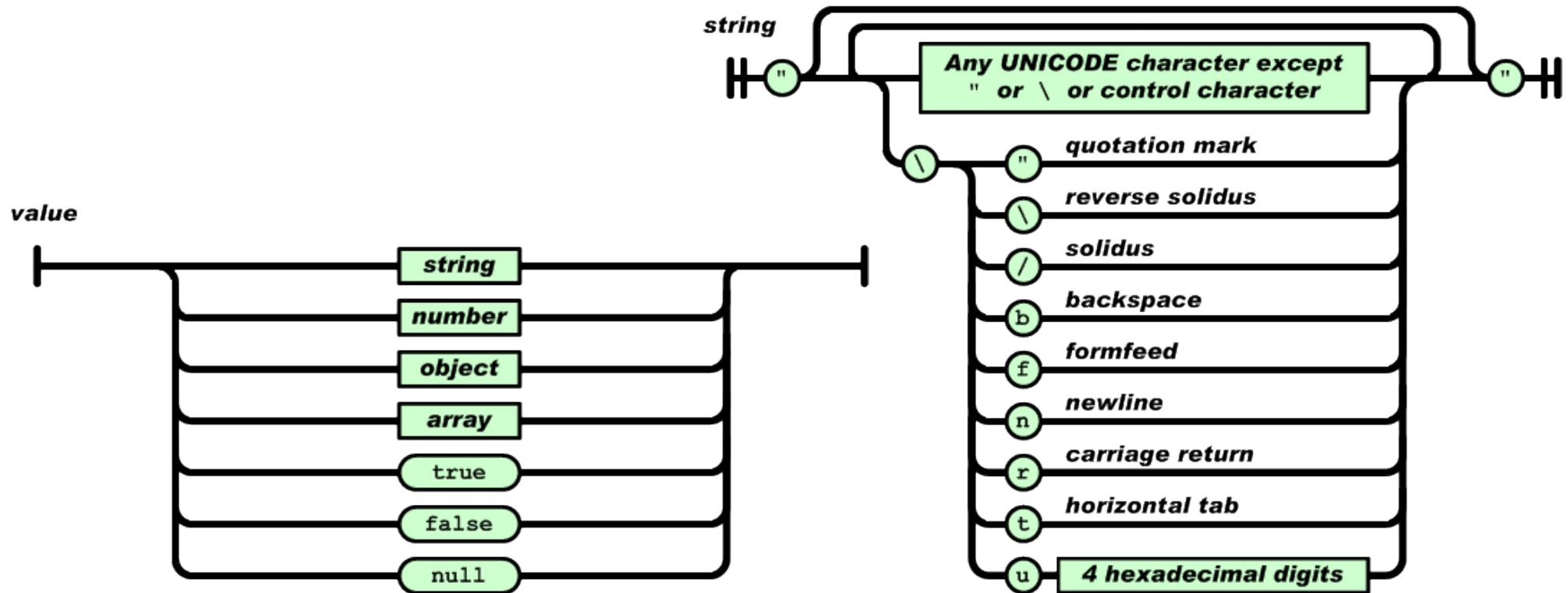
Mária Hatalová
Matej Evin
Simon Nespešný

JSON structure

Java Script Object Notation



JSON structure



JSON structure

```
{ "widget":  
  { "id": 1, "value": "File", "popup":  
    { "menu":  
      [  
        {"value": "New", "onclick": "CreateDoc()"},  
        {"value": "Open", "onclick": null },  
        {"value": "Close", "clickable": false } ]  
      }  
    }  
  }  
}
```

Parser Design

- **C** – personal preference
- **Modular** – functionality split into smaller functions
 - Base function calls functions that parse objects, strings, numbers, etc.
- **Recursive** – JSON structure searched depth-first
- **Emphasis on invalid input** – many error codes

Parser structure

 <code>json_defines.h</code>	patch for empty object
 <code>json_number.c</code>	reworked <code>json_string</code> , added <code>json_number</code>
 <code>json_number.h</code>	reworked <code>json_string</code> , added <code>json_number</code>
 <code>json_parser.c</code>	parser moved from main
 <code>json_parser.h</code>	parser moved from main
 <code>json_string.c</code>	reworked <code>json_string</code> , added <code>json_number</code>
 <code>json_string.h</code>	reworked <code>json_string</code> , added <code>json_number</code>
 <code>main.c</code>	parser moved from main

Parser structure

- Structure **object** – structure to hold the whole JSON
- **main()** is only a dummy
- **json_parse()** is used to get data from JSON
- **parse_object()**, **parse_array()** recursively parse objects/arrays
- **parse_value()** distinguishes value types

Parser structure

- `validate_string()`, `validate_number()` functions check whether strings/numbers are valid
- `find_bracket_pair_object()` and `_array()` functions for finding bracket pairs `{ }` `[]`
- `get_string()`, `free_object_memory()` utility functions

Usage

Download all .c and .h files from git and compile

(we used VisualStudio and CodeBlocks with MinGW, -std=c99)

<https://github.com/hhhj123/PA193-parser/tree/master>

Main function takes file (can be any binary file)

Returns if parsing was successful or not

Structure can be later accessed to print out the whole structure (printing is not implemented)

Team members' competences

Simon Nespešný

- Code design
- Core functionality coding

Matej Evin

- Code design
- Coding of validators
- Code review

Mária Hatalová

- Major code review (security aspects)
- Bug fixes

