

File I/O



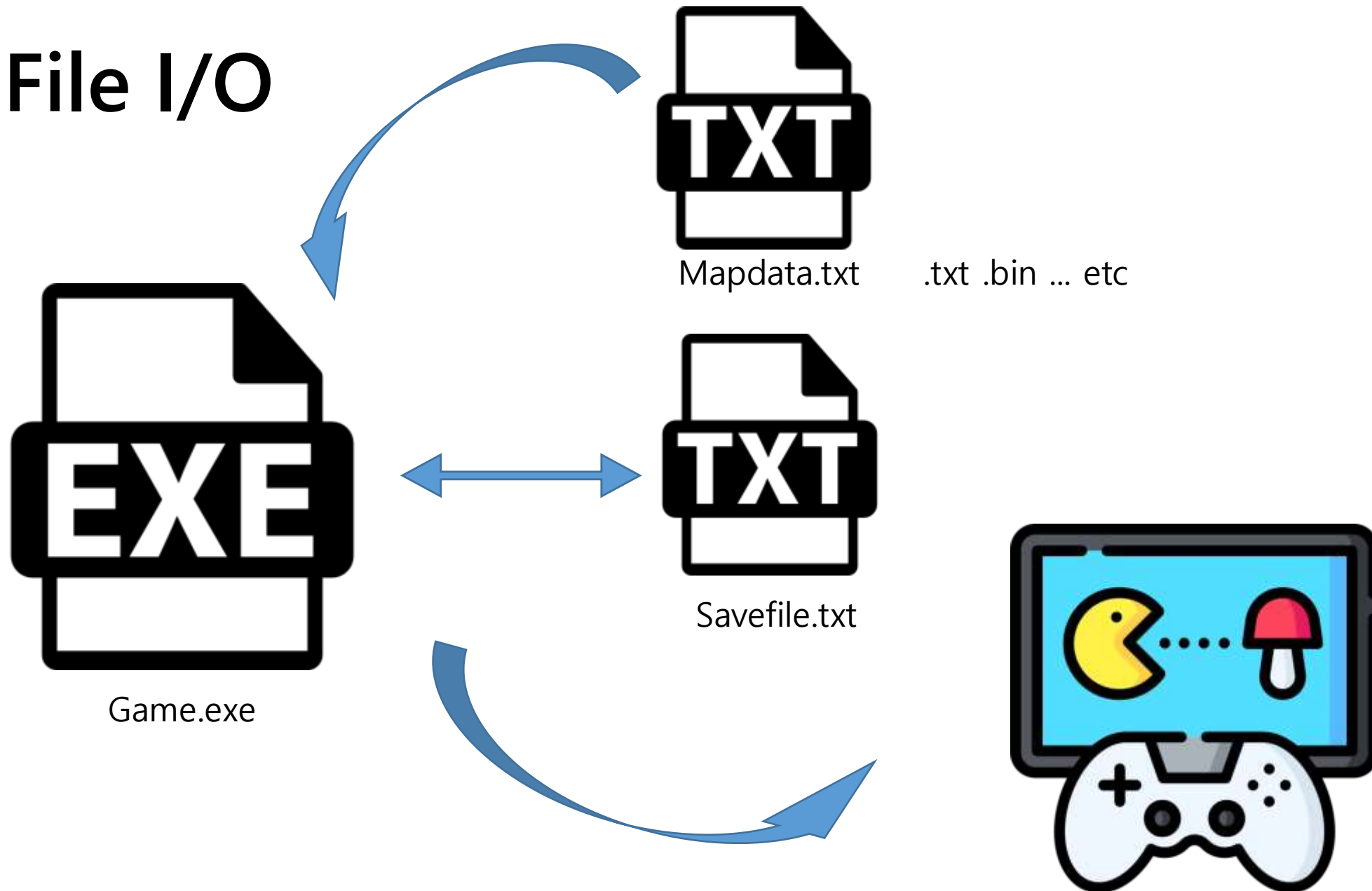
File I/O



Game.exe

To change map and monster's stats after build, we need to change values and re-build the program

File I/O



File I/O

```
#include <stdio.h>
```

```
FILE *fopen(const char * restrict filename, const char * restrict mode);
```

ex)

```
FILE* p_file = fopen("example.txt", "r");
```

```
if(p_file == NULL)
```

```
{
```

```
    // couldn't open file
```

```
}
```

```
else
```

```
{
```

```
    // file opened
```

```
}
```

r: Opens the file for reading only

w: If the file exists already, its contents are overwritten.
If the file doesn't exist, a new file is created

rb: Opens the binary file for reading only

wb: If the file exists already, its contents are overwritten.
If the file doesn't exist, a new binary file is created

<https://www.geeksforgeeks.org/c-fopen-function-with-examples/>

File I/O

```
#include <stdio.h>
```

```
int fclose(FILE *stream);
```

```
ex)
```

```
if ( fclose(p_file) != 0 )
```

```
{
```

```
    // couldn't close file
```

```
}
```

```
else
```

```
{
```

```
    // file closed
```

```
}
```

File I/O

```
#include <stdio.h>

int main()
{
    FILE* p_file = fopen("example.txt", "r");
    if(p_file == NULL)
    {
        // couldn't open file
    }
    else
    {
        // file opened
        // do something here
        if ( fclose(p_file) != 0 )
        {
            // couldn't close file
        }
        else
        {
            // file closed
        }
    }

    return 0;
}
```

File write

```
#include <stdio.h>

int main()
{
    FILE *p_file = fopen("data.bin", "wb");
    if (!p_file)
    {
        printf("Couldn't open file.\n");
        return 1;
    }

    char s[15]="hello world!";

    fwrite(s, sizeof(s), 1, p_file);

    fclose(p_file);

    return 0;
}
```

test.c

data.bin

6865 6c6c 6f20 776f 726c 6421 0000 00

fputc(): Write a single character

<https://en.cppreference.com/w/c/io/fputc>

fputs(): Write a string

<https://en.cppreference.com/w/c/io/fputs>

fprintf(): Write formatted text

<https://en.cppreference.com/w/c/io/fprintf>

fwrite(): Write binary data

<https://en.cppreference.com/w/c/io/fwrite>

File read

```
#include <stdio.h>

int main()
{
    FILE *p_file = fopen("data.bin", "rb");

    if (!p_file)
    {
        printf("Coudln't open file.\n");
        return 1;
    }

    char s[15] = {0};

    fread(s, sizeof(char), sizeof(s)-1, p_file);
    s[sizeof(s)-1]='\0';

    printf("%s\n",s);
    fclose(p_file);

    return 0;
}

jinwoo@DESKTOP-UEN32NR: ~$ ./a.out
hello world!
```

fgetc(): Read by character

<https://en.cppreference.com/w/c/io/fgetc>

fgets(): Read by line

<https://en.cppreference.com/w/c/io/fgets>

fread(): Read binary data

<https://en.cppreference.com/w/c/io/fread>

JSON(JavaScript Object Notation)

- JSON is a widely used format for data storage and exchange.
- C doesn't natively support JSON but we can use external libraries
- JSON is commonly used in game data, configuration files, APIs, and network protocols

```
{
  "DocumentType": 1,
  "No.": "S-ORD101001",
  "SellToCustNo": "10000",
  "PostingDate": "2023-04-02",
  "Lines": [
    {
      "LineNo": 10000,
      "Type": 2,
      "No": "1996-S",
      "Quantity": 12,
      "UnitPrice": 1397.3
    },
    {
      "LineNo": 20000,
      "Type": 2,
      "No": "1900-S",
      "Quantity": 4,
      "UnitPrice": 192.8
    }
  ]
}
```

.json file

LAB – FileIO

- Create a file named 'FileIO_YourName.c'.
- Store a predefined password into a file named **password.txt**
- Continuously prompt the user to enter a password using a **while loop**
- **Read the stored password from password.txt** and compare it with the user input
- If the entered password is correct, display a success message and terminate the program
- **If the password is incorrect, prompt the user to try again**