

Group_7 Assignment_2 report

Specifications

Write a program function like **cp(1)** which copies the file. In your implementation, you should implement **your own cp(1)** function **without directly using the cp(1) or any other similar command**. The following is the grading policy:

- (1) (3 pt) Implement **your own cp(1)** which can copy the file correctly.
- (2) (1 pt) Describe your implementation in your report.

- Below is the output after we run the command `diff source.txt dest.txt`



- Below is the description of our implementation

```
#include <stdio.h>
#include <stdlib.h>
#define BUFFER_SIZE 4096
```

- We define the buffer size `BUFFER_SIZE` as 4096, This will be the size of the buffer used for reading and writing files.

```

void copy_file(const char *source, const char *destination) {
    FILE *src_file = fopen(source, "rb");
    FILE *dest_file = fopen(destination, "wb");
    if (src_file == NULL || dest_file == NULL) {
        printf("Error: Unable to open file.\n");
        exit(1);
    }

    char buffer[BUFFER_SIZE];
    size_t bytes_read;
    while ((bytes_read = fread(buffer, sizeof(char), BUFFER_SIZE, src_file)) > 0) {
        fwrite(buffer, 1, bytes_read, dest_file);
    }

    fclose(src_file);
    fclose(dest_file);
}

```

- First, the code will use `fopen` to open the source file and destination file in read-binary mode and write-binary mode, we will do the check to ensure that both source file and the destination file had been successfully open.
- After open the both file, we will use a buffer of `BUFFER_SIZE` bytes to read from the source file and write to the destination file.
 - For the file reading and file writing, we choose to use the function `fread` and `fwrite` to write the content from the source file to the destination file.
- Finally, we will use the function `fclose` to close the source file and destination file.

```

int main(int argc, char *argv[]) {
    if (argc != 3) {
        printf("Usage: %s <source> <destination>\n", argv[0]);
        return 1;
    }

    copy_file(argv[1], argv[2]);
    return 0;
}

```

- For the main function, we first check the input is correct, and call the function `copy_file` to copy the data from source file to destination file.