File Handling

Aim – Write a program to read, write and append and file copy operations.
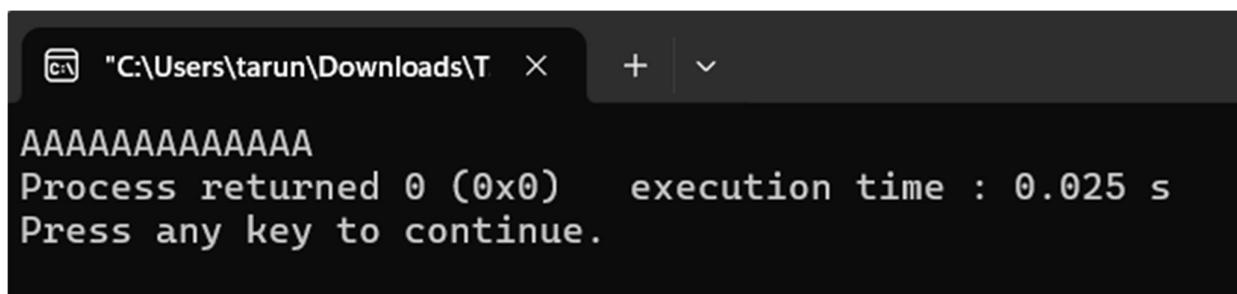
Theory –
File handling in C allows programs to store and retrieve data using external text files. The function `fopen()` opens a file in different modes such as read, write, or append. Characters are read from a file using `fgetc()`, and written using `fputc()`. To display characters on the screen, `putchar()` is used. User input is accepted through `scanf()`, which helps collect characters to write into the file. After all operations are completed, `fclose()` is used to close the file, ensuring data is properly saved and resources are released. These functions make text processing efficient and structured.

A1.

```c
#include <stdio.h>

void main() {
    FILE *fp;
    char ch;
    fp = fopen("tsec.txt", "r");
    while ((ch = fgetc(fp)) != -1) {
        putchar(ch);
    }
    fclose(fp);
}
```



```
AAAAAAAAAAAAA
Process returned 0 (0x0)    execution time : 0.025 s
Press any key to continue.
```

A2.
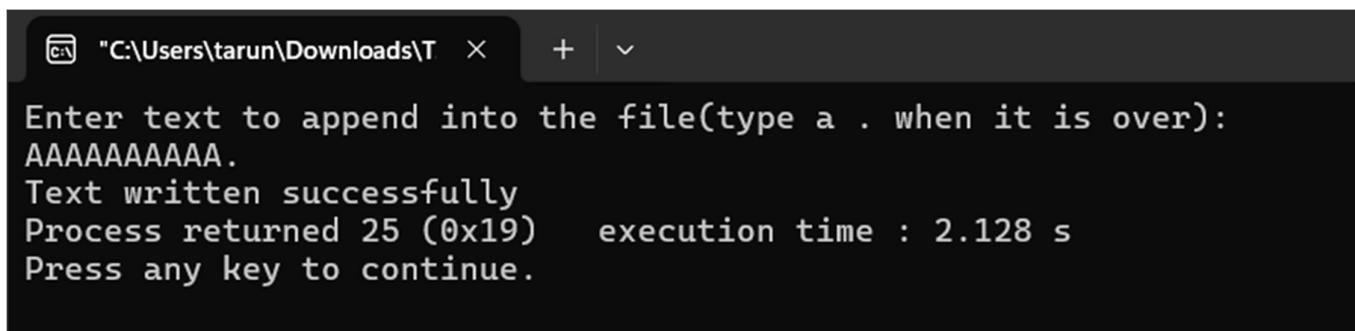
```c
#include <stdio.h>
void main() {
    FILE *fp;
    fp = fopen("tsec.txt", "w");
    printf("Enter text to append into the file(type a . when it is over):\n");
    char ch;
    while (ch != '.'){
```

```
        fputc(ch, fp);

        scanf("%c", &ch);

    }

    fclose(fp);

    printf("Text written successfully");

}
```
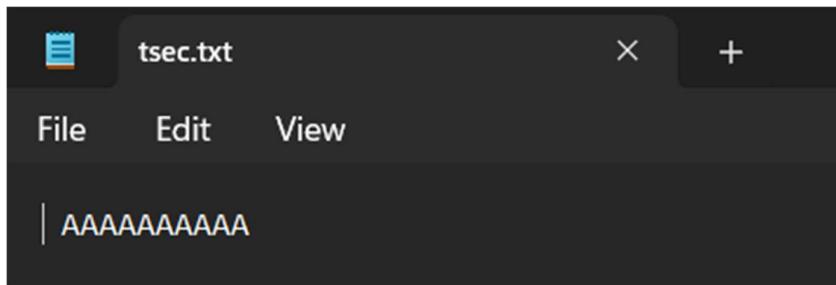
```
Cn  "C:\Users\tarun\Downloads\T   ✕     +   ∨

Enter text to append into the file(type a . when it is over):
AAAAAAAAAA.
Text written successfully
Process returned 25 (0x19)    execution time : 2.128 s
Press any key to continue.
```

```
📄  tsec.txt                              ✕     +

File    Edit    View

| AAAAAAAAAA
```

A3.
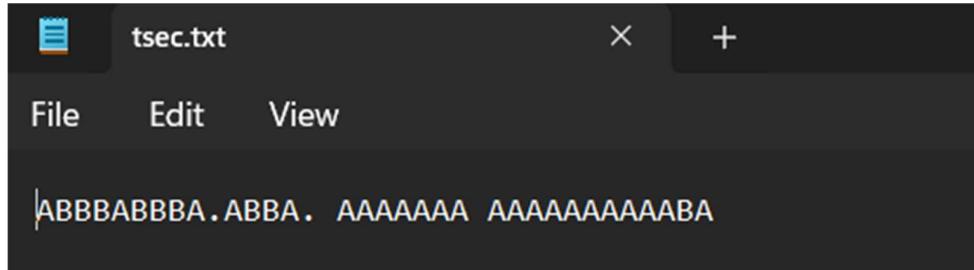
```
#include <stdio.h>

void main() {

    FILE *fp;

    fp = fopen("tsec.txt", "a");

    printf("Enter text to append into the file(type a . when it is over):\n");

    char ch;

    while (ch != '.'){

        fputc(ch, fp);

        scanf("%c", &ch);

    }

    fclose(fp);

    printf("Text written successfully");

}
```

```
Enter text to append into the file(type a . when it is over):
AAAAAAAAAABA.
Text written successfully
Process returned 25 (0x19)   execution time : 3.183 s
Press any key to continue.
```

tsec.txt

File   Edit   View

ABBBABBBA.ABBA. AAAAAAA AAAAAAAAAABA
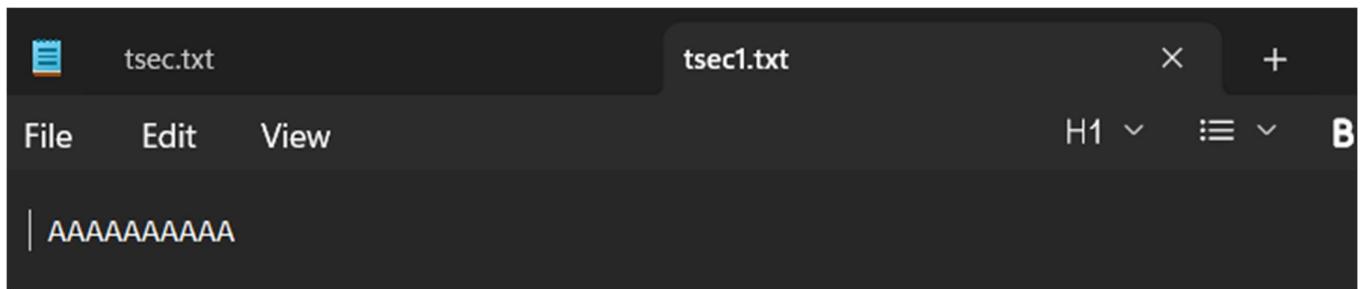
A4.

#include <stdio.h>

void main() {

   FILE *source, *dest;

   char ch;

   source = fopen("tsec.txt", "r");

   dest = fopen("tsec1.txt", "w");

   while ((ch = fgetc(source)) != -1) {

      fputc(ch, dest);

   }

   printf("File copied successfully!\n");

   fclose(source);

   fclose(dest);

}

```
File copied successfully!

Process returned 0 (0x0)   execution time : 0.024 s
Press any key to continue.
```

Conclusion

I learned how to open, read, write, append, and close files using C file-handling functions. These operations let programs store data externally and process text efficiently and safely.