Experiment 11

Absolute Loader

Surya Narayanan
s5 csb 48

Step 1: START
Step 2: Include necessary header files
Step 3: Assign the required variables
Step 4: Open the files

```
fp1 = fopen ("input.txt", "r");
fp2 = fopen ("output.txt", "w")
```

Step 5: Read the content of input file
Step 6: using while loop perform the loop until character is not equal to E

Step 6.1 - Compare weather the character is equal to H.

Step 6.2 - If type is H then get the starting address, length & speed.

Step 6.3 - a): otherwise, if the character is T, then store the string as the three address in the output file with
input[0], input[1] for address
input[2], input[3] for address+1
input[4], input[5] for address+2

b: increment the address value by 3

c: Read the next input string.

STEP 6.4: a- Otherwise if it's not it' O T then perform the following print statements in fp2 for outputfile for:

Input[0], input[1] for address
input[4], input[5] for address +1
input[5] for address +2.

b: increment in address value by 3

c: Read the next input string

STEP 6.5 : to Repeat from step 5

STEP 7: close files

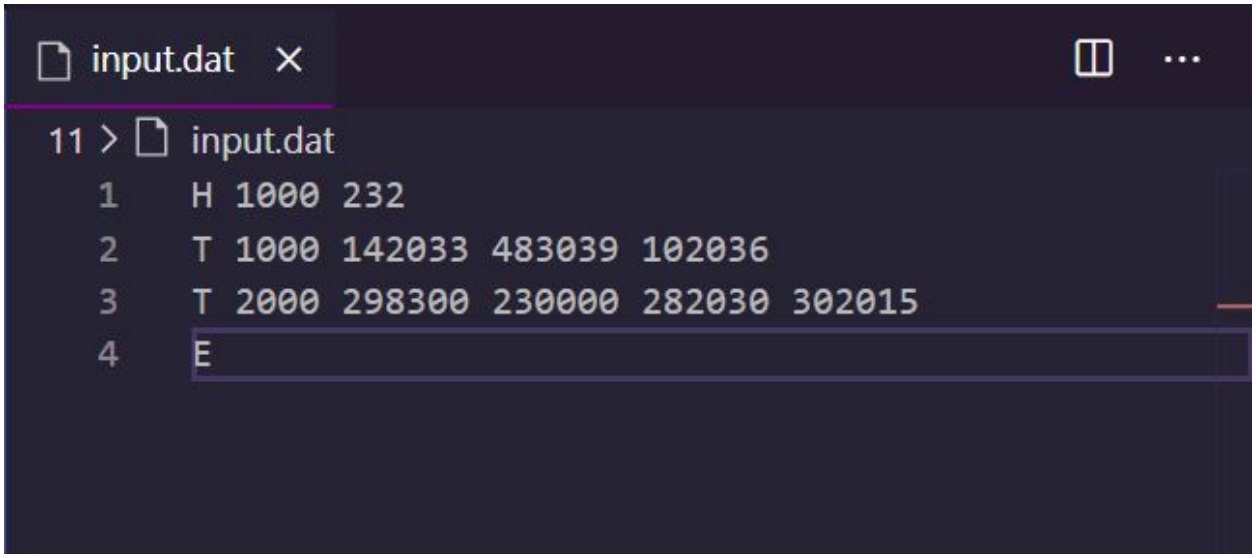
STEP 8 : Display "finished"

Program

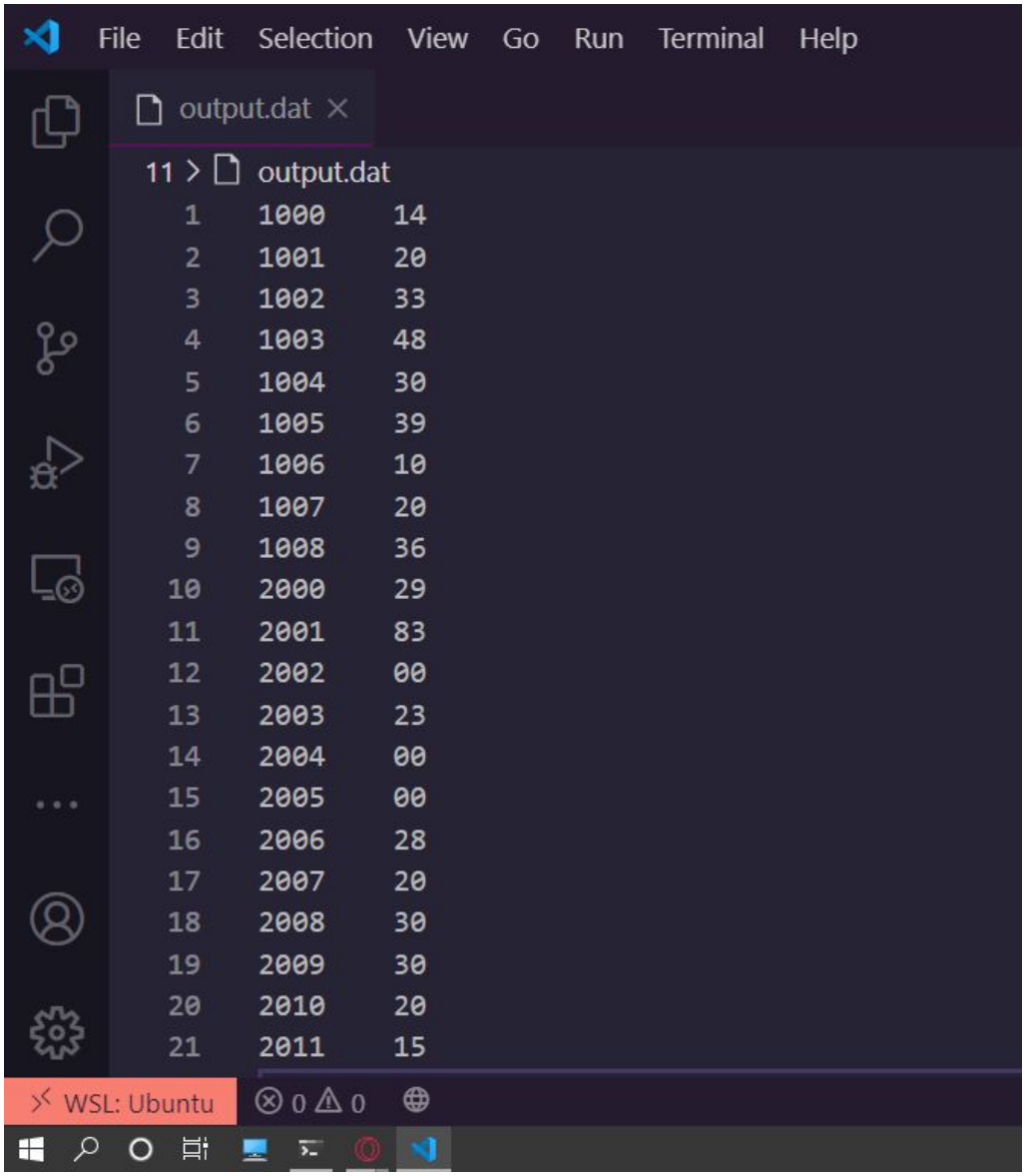
```
#include <stdio.h>
#include <conio.h>
#include <string.h>
void main()
{
    char input[10];

    int start, length, address;
    FILE *fp1, *fp2;

    fp1 = fopen("input.dat", "r");
    fp2 = fopen("output.dat", "w");
    fscanf(fp1, "%s", input);

    while (strcmp(input, "E") != 0)
    {
        if (strcmp(input, "H") == 0)
        {
            fscanf(fp1, "%d", &start);
            fscanf(fp1, "%d", &length);
            fscanf(fp1, "%s", input);
        }

        else if (strcmp(input, "T") == 0)
        {
            fscanf(fp1, "%d", &address);
            fscanf(fp1, "%s", input);
            fprintf(fp2, "%d\t%c%c\n", address, input[0], input[1]);
            fprintf(fp2, "%d\t%c%c\n", (address + 1), input[2], input[3]);
            fprintf(fp2, "%d\t%c%c\n", (address + 2), input[4], input[5]);
            address += 3;

            fscanf(fp1, "%s", input);
        }
        else
        {
            fprintf(fp2, "%d\t%c%c\n", address, input[0], input[1]);
            fprintf(fp2, "%d\t%c%c\n", (address + 1), input[2], input[3]);
            fprintf(fp2, "%d\t%c%c\n", (address + 2), input[4], input[5]);
```

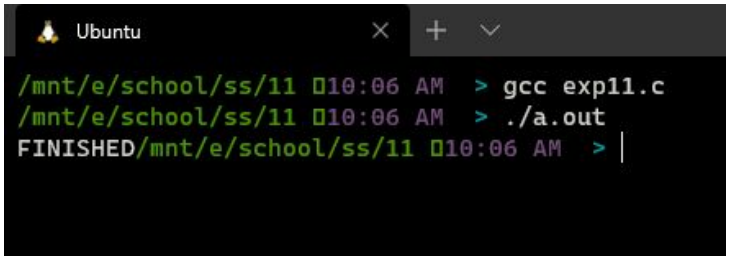
```
            address += 3;
            fscanf(fp1, "%s", input);
        }
    }
    fclose(fp1);
    fclose(fp2);
    printf("FINISHED");
}
```

Input



Output


File Edit Selection View Go Run Terminal Help
output.dat
11 > output.dat
1 1000 14
2 1001 20
3 1002 33
4 1003 48
5 1004 30
6 1005 39
7 1006 10
8 1007 20
9 1008 36
10 2000 29
11 2001 83
12 2002 00
13 2003 23
14 2004 00
15 2005 00
16 2006 28
17 2007 20
18 2008 30
19 2009 30
20 2010 20
21 2011 15
WSL: Ubuntu
0 0


Ubuntu
/mnt/e/school/ss/11 10:06 AM > gcc exp11.c
/mnt/e/school/ss/11 10:06 AM > ./a.out
FINISHED/mnt/e/school/ss/11 10:06 AM >