ARTI BHATIA

CS5A, 16

**PASS 1 ASSEMBLER**

Q- Implement pass one of a two pass assembler.

Program:

/\*

  Implementation of pass one of a two pass assembler.

\*/

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

int main(){

    int start, locctr, length;

    char opcode[10], operand[10], label[10], code[10], mnemonic[10];

    FILE \*fp1, \*fp2, \*fp3, \*fp4;

    fp1 = fopen("Input.txt", "r");

    fp2 = fopen("OPTAB.txt", "r");

    fp3 = fopen("SYMTAB.txt", "w");

    fp4 = fopen("Output.txt", "w");

    fscanf(fp1,"%s\t%s\t%s", label, opcode, operand);

    if(strcmp(opcode,"START") == 0){

        start = atoi(operand);

        locctr = start;

        fprintf(fp4,"\t%s\t%s\t%s\n", label, opcode, operand);

        fscanf(fp1,"%s\t%s\t%s", label, opcode, operand);

    }

    else

        locctr = 0;

    while(strcmp(opcode,"END") != 0){

        fprintf(fp4, "%d\t", locctr);

        if(strcmp(label,"\*\*") != 0)

            fprintf(fp3,"%s\t%d\n", label, locctr);

        fscanf(fp2, "%s\t%s", code, mnemonic);

        while(strcmp(code,"END") !=0 ){

            if(strcmp(opcode,code) == 0){

                locctr += 3;

                break;

            }

            fscanf(fp2, "%s\t%s", code, mnemonic);

        }

        if(strcmp(opcode, "WORD") == 0)

            locctr += 3;

        else if(strcmp(opcode, "RESW") == 0)

            locctr += 3\*(atoi(operand));

        else if(strcmp(opcode, "RESB") == 0)

            locctr += atoi(operand);

        else if(strcmp(opcode, "BYTE")==0)

            ++locctr;

        fprintf(fp4, "%s\t%s\t%s\t\n", label, opcode, operand);

        fscanf(fp1, "%s\t%s\t%s", label, opcode, operand);

    }

    fprintf(fp4, "%d\t%s\t%s\t%s\n", locctr, label, opcode, operand);

    length = locctr - start;

    printf("\nLength of the code : %d\n",length);

    fclose(fp1);

    fclose(fp2);

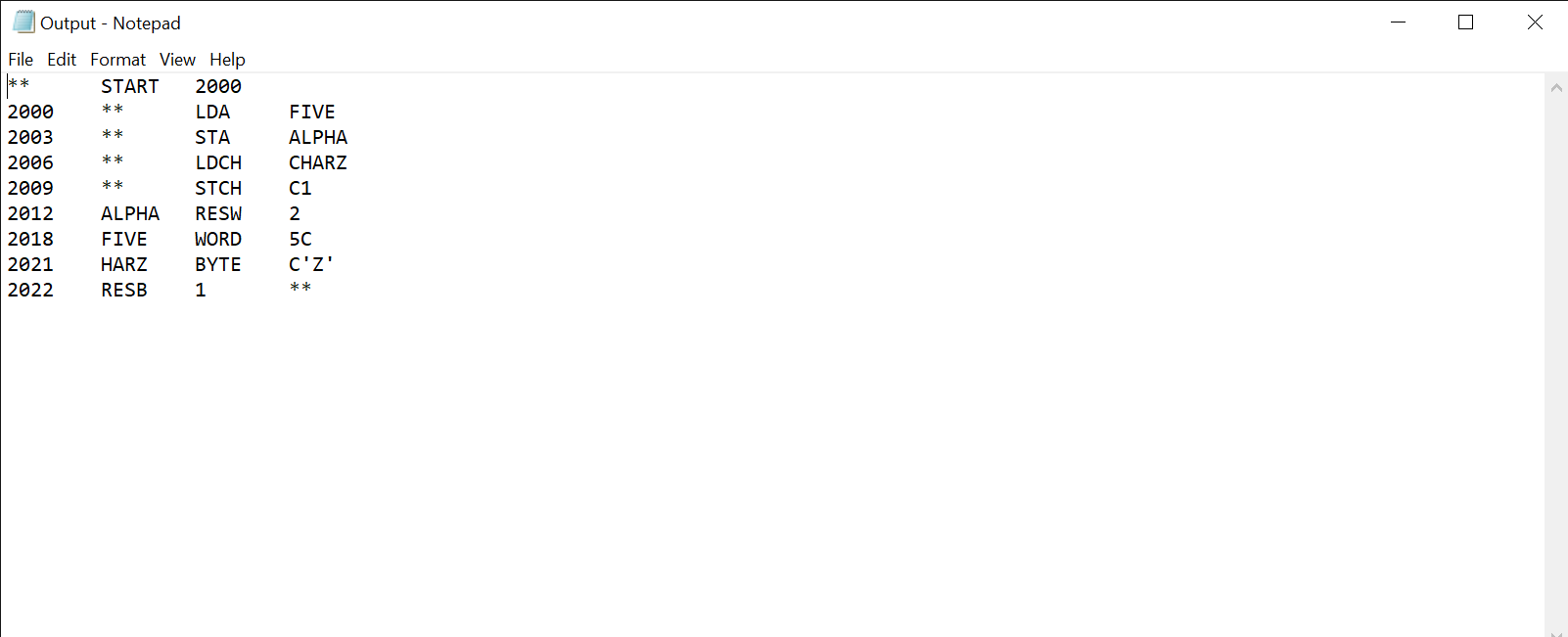
    fclose(fp3);

    fclose(fp4);

    return 0;

}

Output.txt File:



Algorithm:

