EXPERIMENT-13

Implement a single pass assembler

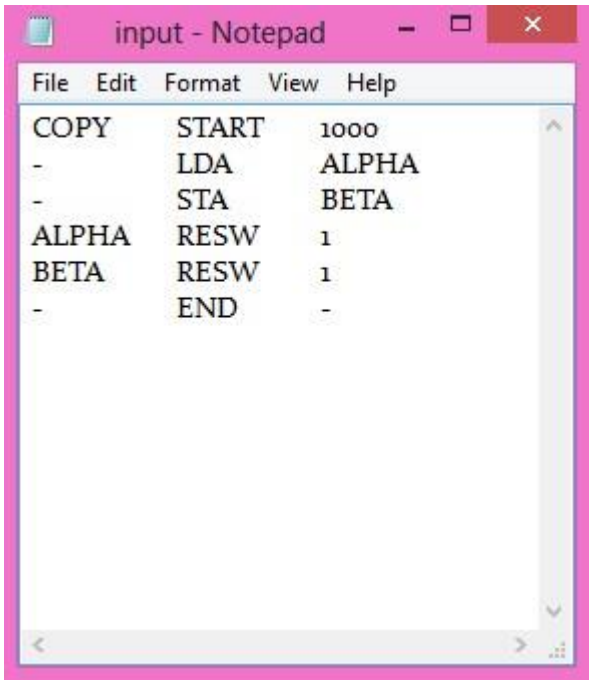
Source Code

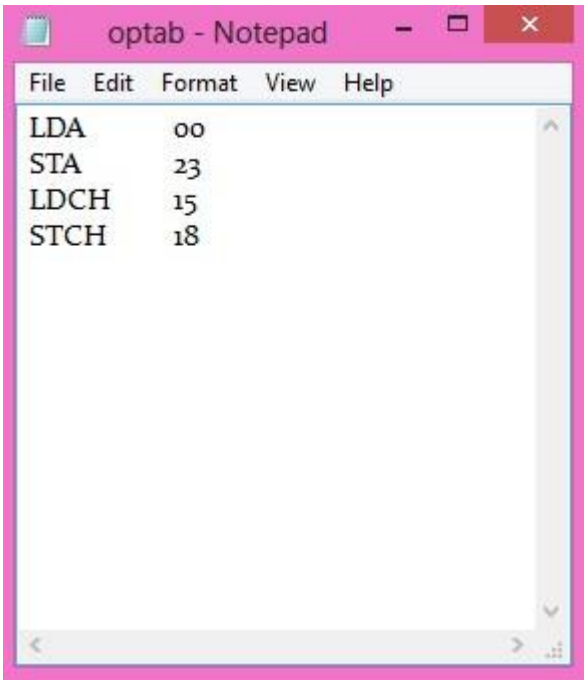
```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
#include<string.h>

void main() {
  FILE * f1, * f2, * f3, * f4, * f5;   int lc, sa, i = 0, j = 0, m[10], pgmlen, len, k,
len1, l = 0;   char name[10], opnd[10], la[10], mne[10], s1[10], mne1[10],
opnd1[10],lcs[10], ms[10];   char sym[10], symaddr[10], obj1[10], obj2[10], s2[10],
q[10], s3[10];   f1 = fopen("input.txt", "r");   f2 = fopen("optab.txt", "r");   f3 =
fopen("symtab.txt", "w+");   f4 = fopen("symtab1.txt", "w+");   f5 =
fopen("output.txt", "w+");   fscanf(f1, "%s%s%s", la, mne, opnd);   if (strcmp(mne,
"START") == 0) {     sa = atoi(opnd);     strcpy(name, la);     lc = sa;
  }   strcpy(s1, "*");   fscanf(f1,
"%s%s%s", la, mne, opnd);   while
(strcmp(mne, "END") != 0) {     if
(strcmp(la, "-") == 0) {
fscanf(f2, "%s%s", mne1, opnd1);
while (!feof(f2)) {         if
(strcmp(mne1, mne) == 0) {
m[i] = lc + 1;
          fprintf(f3, "%s\t%s\n", opnd,
s1);           fprintf(f5, "%s\t0000\n",
opnd1);           lc = lc + 3;           i
= i + 1;           break;         } else
fscanf(f2, "%s%s", mne1, opnd1);
      }

    } else {       fseek(f3, SEEK_SET,
0);       fscanf(f3, "%s%s", sym,
symaddr);
```

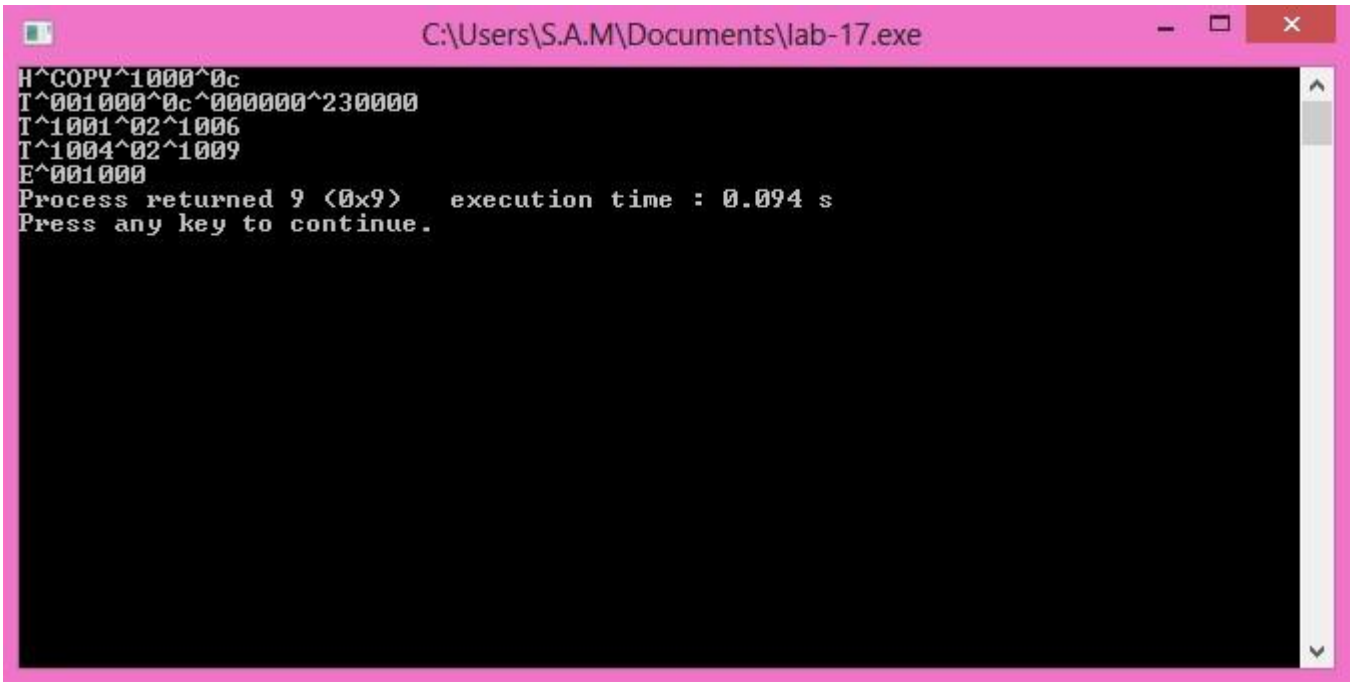
```
      while (!feof(f3)) {         if
(strcmp(sym, la) == 0) {           itoa(lc,
lcs, 10);           fprintf(f4, "%s\t%s\n",
la, lcs);           itoa(m[j], ms, 10);
j = j + 1;           fprintf(f5,
"%s\t%s\n", ms, lcs);           i = i + 1;
break;         } else           fscanf(f3,
"%s%s", sym, symaddr);
      }       if (strcmp(mne, "RESW") ==
0)         lc = lc + 3 * atoi(opnd);
else if (strcmp(mne, "BYTE") == 0) {
strcpy(s2, "-");         len =
strlen(opnd);         lc = lc + len - 2;
for (k = 2; k < len; k++) {
q[l] = opnd[k];           l = l + 1;
        }         fprintf(f5,
"%s\t%s\n", q, s2);         break;
      } else if (strcmp(mne, "RESB") ==
0)         lc = lc + atoi(opnd);
else if (strcmp(mne, "WORD") == 0) {
strcpy(s3, "#");         lc = lc + 3;
fprintf(f5, "%s\t%s\n", opnd, s3);
break;
      }
    }
    fseek(f2, SEEK_SET, 0);
fscanf(f1, "%s%s%s", la, mne, opnd);
  }
  fseek(f5, SEEK_SET, 0);   pgmlen = lc -
sa;   printf("H^%s^%d^0%x\n", name, sa,
pgmlen);   printf("T^");
  printf("00%d^0%x", sa, pgmlen);
fscanf(f5, "%s%s", obj1, obj2);   while
(!feof(f5)) {
    if (strcmp(obj2, "0000") == 0)
printf("^%s%s", obj1, obj2);     else
if (strcmp(obj2, "-") == 0) {
printf("^");       len1 =
strlen(obj1);       for (k = 0; k <
len1; k++)
```

```
        printf("%d", obj1[k]);
    } else if (strcmp(obj2, "#") == 0)
{       printf("^");       printf("%s",
obj1);
    }     fscanf(f5, "%s%s", obj1,
obj2);
  }   fseek(f5, SEEK_SET, 0);   fscanf(f5,
"%s%s", obj1, obj2);   while (!feof(f5)) {
if (strcmp(obj2, "0000") != 0) {       if
(strcmp(obj2, "-") != 0) {         if
(strcmp(obj2, "#") != 0) {
printf("\n");
printf("T^%s^02^%s", obj1, obj2);         }
      }     }     fscanf(f5,
"%s%s", obj1, obj2);
  }   printf("\nE^00%d",
sa);
}
```

Output







SUBMITTED BY:

NIVEA GIGEN

S5-C

43

CHN18CS092