

```

1 #include "simulator.h"
2
3 void load(char * target, void * memory, int * program_begin, int * program_end, int
* loaded)
4 {
5     char buffer[100];
6     FILE * load_obj = fopen(target, "r");
7     if(load_obj == NULL)
8     {
9         printf("error\n");
10    }
11    else
12    {
13        *loaded = 1;
14        char begin[20];
15        int size = 0;
16        while(fgets(buffer, 100, load_obj))
17        {
18            if(*buffer == 'H')
19            {
20                for(int i = 0 ; i < 6 ; i++)
21                {
22                    begin[i] = buffer[i + 7];
23                }
24                begin[6] = '\0';
25                printf("%d ", hex_to_dex(begin));
26                *program_begin = hex_to_dex(begin);
27
28                size = hex_to_dex(buffer+13);
29                printf("%d \n", size);
30                *program_end = *program_begin + size;
31            }
32
33            if(*buffer == 'T')
34            {
35                for(int i = 0 ; i < 6 ; i++)
36                {
37                    begin[i] = buffer[i + 1];
38                }
39                begin[6] = '\0';
40
41                printf("%X ", hex_to_dex(begin));
42                int memory_count = hex_to_dex(begin);
43                int state = 0;
44
45                for(int i = 9 ; i < strlen(buffer) - 1 ; i++)
46                {
47                    if(state == 0)
48                    {
49                        *((uint8_t *)memory + memory_count) =
50(hex_to_dex_c(buffer[i]) << 4) ;
51                        printf("%c", buffer[i]);
52                        state = 1;
53                    }
54                    else if(state == 1)
55                    {
56                        *((uint8_t *)memory + memory_count) |=
57(hex_to_dex_c(buffer[i]) << 0) ;
58                        memory_count++;
59                        printf("%c", buffer[i]);
60                        state = 0;
61                    }
62                }
63                printf("\n");
64            }
65        }
66        fclose(load_obj);
67    }
68 }

```