Derek Nola

CPE 329

Assignment 0

6 Bit Counter:

```c
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char** argv){

    int i = 0;
    printf("D5 D4 D3 D2 D1 D0\n");
    printf("-----------------\n");

    while(i < 64){
        printf("%x  %x  %x  %x  %x  %x\n",
            (i & 32) >> 5, (i & 16) >> 4, (i & 8) >> 3,
            (i & 4) >> 2, (i & 2) >> 1, i & 1);
        i++;
    }

    return 0;
}
```

Word Sort:

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

#define WORD_LEN 256

int cmpStr(const void* a, const void* b){
    return strcmp(*(const char**) a, *(const char**)b);
}

void printList(char** list, int size){
    int i;
    printf("Sorted List:\n");
    for(i = 0; i < size; i++)
        printf("%s", list[i]);
}
```

```c
int main(int argc, char** argv){
    FILE* file = NULL;
    char* fileName = (char*) malloc(WORD_LEN);
    printf("Enter the name of the file to sort: ");
    scanf("%s", fileName);
    if(NULL == (file = fopen(fileName, "r+"))){
        fprintf(stderr, "Failed to open %s file\n", fileName);
        exit(1);
    }

    char length[WORD_LEN];
    fgets(length, WORD_LEN, file);
    int len_of_file = atoi(length);
    char** list = malloc(sizeof(char*) * len_of_file);
    int i;
    for(i = 0; i < len_of_file + 1; i++){
        list[i] = (char*) malloc(WORD_LEN);
        fgets(list[i], WORD_LEN, file);
    }
    qsort(list, i - 1, sizeof(char*), cmpStr);
    printList(list, i);
    return 0;
}
```