

```
/* Räkneövning 1, Uppgift 1*/
#include<stdio.h>
#include<stdlib.h>

#define MAX_LEN 254

int main (int argc, char *argv[]) {
    char text[MAX_LEN];
    char *ptr = NULL;

    ptr = fgets(/* Ange rätt parametrar */);

    if (ptr == NULL) {

        printf("An error occurred");
        exit (EXIT_FAILURE);
    }
    else {
        /*printa ut det du skrivit in från stdin*/
    }

    return (0);
}
```

```

/* Räkneövning 1, Uppgift 2 */

//inkludera biblioteket för input/output
//inkludera biblioteket för stränghantering
//inkludera biblioteket för användning av exit(), atoi, m.fl.

/*Structen*/
typedef struct address {
    ...
}address;

/*Prototyper*/
void printText(struct address *adr, FILE *p);

int main(int argc, char *argv[]) {
    address adr;
    FILE *fileDB = NULL;
    fileDB = fopen (argv[1], "w");
    if (fileDB == NULL) {
        /* Avbryt programmet ifall fileDB pekar på NULL*/
    } else {
        /* Mata in uppgifterna till structen */
    }
    /*Anropa funktionen printText med structen
och en file stream som param.*/
    printText(&adr, fileDB);
    /* Kom ihåg att stänga filströmmen! */
    fclose (fileDB);
    return 0;
}

void printText(struct address *adr, FILE *file) {

    // Checka att file inte pekar på NULL
    if ( file == NULL) {
        exit(EXIT_FAILURE);
    } else {
        /*Skriv till stdout eller till fil genom att t.ex skriva:*/
        fprintf(file, "Namn: %s %s\n", adr->forNamn,
                adr->efterNamn);
        ....
    }
}

```

```

/* Räkneövning 1, Uppgift 3*/

//inkludera biblioteket för input/output
//inkludera biblioteket för stränghantering
//inkludera biblioteket för användning av exit(), atoi, m.fl.

/*Structen*/
typedef struct address {}address;

struct nod {};

/* Global pekare till listhuvud */
struct nod *listHead = NULL;

/* Prototyper */
int readData(struct address *);
int newAddressToList();
int writeAddress (FILE * );
int emptyList ();

int main(int argc, char *argv[]) {
    address adr;
    FILE *fileDB = NULL;
    if (fileDB pekar på null) {
        /* Avbryt programmet ifall fileDB pekar på NULL*/
    } else {
        /* Lägg till minst 3 element till den länkade listan */
        int i;
        for (i=0; i < 3; i++) { newAddressToList(); }

        /* Skriv sedan listan till stdout och filen */
        writeAddress (fileDB);
        writeAddress (stdout);

        /* Töm till slut den länkade listan och stäng filströmmen */
        emptyList();
        fclose (fileDB);
        return 0;
    }
}

int readData(struct address * addr) {...}
int newAddressToList() {...}
int writeAddress (FILE * fil) {...}
int emptyList () {...}

```