

```

// usual includes
#include <stdio.h>
#include <stdlib.h>
#include <strings.h>

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

    //Various printing methods
    printf("String : %s", stringVar);
    printf("Character : %c", charVar);
    printf("Decimal : %d", decimalVar);

    -----

    //Get the length of a string
    char[50] str;
    int length;

    strcpy(str, "This is the string I'm testing");
    length = strlen(str);

    -----

    //Compare two strings without case sensitivity

    char *str1 = "STRING ONE";
    char *str2 = "string TWO";
    int result;

    result = strncasecmp(str1, str2, 10);

    //Result < 0 -> string1 less than string2
    //Result > 0 -> string1 greater than string2
    //Result = 0 -> string1 equivalent to string2

    -----

    //Find a substring within a string

    char* modeNum;
    int mode;
    modeNum = argv[1];

    if(strcasestr("encryption", modeNum))
        mode = 0;

    else if(strcasestr("decryption", modeNum))
        mode = 1;

    -----

    //Converting a string (char array) to an int

    int val;
    char str[20];

    strcpy(str, "98993489");
    val = atoi(str);

    -----

    //Copying part of a buffer to another buffer
    memcpy(coinbuf, &value, STANDARDLENGTH);
    memcpy(coinbuf+4, &newbits, STANDARDLENGTH);
    memcpy(coinbuf+8, &newkeylen, STANDARDLENGTH);

    -----

    //Copy command line args to a string
    FILE *fptr;
    char filename[PATH_MAX];

    strncpy(filename,argv[1],PATH_MAX);

    //Can also do it from user input

```

```

printf("Enter the filename to open \n");
char *rv = fgets(filename,PATH_MAX,stdin);

-----

//Opening a file

    fptr = fopen(filename, "r");

    if (fptr == NULL) {
        printf("Cannot open file: %s\n",filename);
        exit(5);
    }

-----

// Reading contents from a file

char c;
fptr = fopen(filename, "r");

//Gets char at pointer
c = (char) fgetc(fptr);

//While c != EndOfFile
while (c != EOF) {

    //Prevents binary crap being printed
    if (!isprint(c) && !isspace(c) && c!='\n' && c!='\r') {
        printf ("0x%x", c);
    } else {
        printf ("%c", c);
    }
    c = fgetc(fptr);
    printf ("%c", c);
}

    fclose(fptr); // close the file

}

-----

int validchar(char ch)
{
    // we know the ascii table has these continuities
    if (ch >= 'A' && ch<='Z') return(0);
    if (ch >= 'a' && ch<='z') return(0);
    if (ch >= '0' && ch<='9') return(0);
    if (ch=='.' || ch=='-' || ch=='_') return(0);
    return(1);
}

-----

int checkDNSValid(char *str)
{
    // chars valid?
    for (char *ch=str;*ch;ch++)
        if (validchar(*ch))
            return(3);
}

-----

void usage(char *prog)
{
    fprintf(stderr,"usage: %s [list-of-files]\n",prog);
    exit(1);
}

int main(int argc,char *argv[])
{
    // read args
    if (argc<2)

```

```
usage(argv[0]);  
}
```