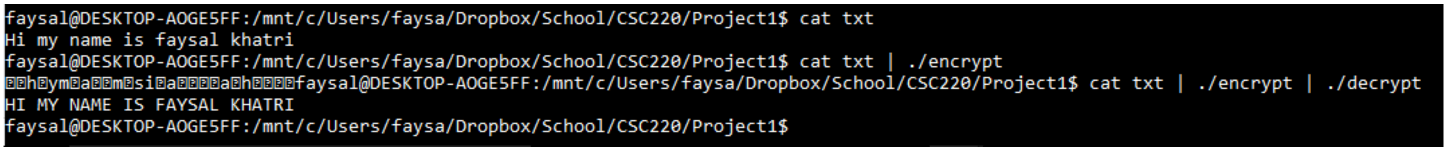
Faysal Khatri
CSC220 -- Project 1
2017-06-12

Design Notes

1. The encrypt program uses scanf to read in a string from stdin. Using this expression, I was able to get it to read in until it reached a new line:
   `scanf("%[^\n]s", s);`
2. Only spaces and letters A-Z are allowed. If other characters are detected, the program provides an error message and stops.
3. A key is hard coded and used to encrypt the inputted string until the key is exhausted, then the string itself acts as the key.
4. The encrypted characters are stored in an unsigned char array and finally printed out.
5. The decrypt program reads in an unsigned char array from standard in. It contains the same hard coded key and follows the same process to decrypt.
6. Since only spaces and A-Z are allowed, the smallest encrypted value is 64 (space + space) and the greatest is 180 (Z + Z). Decrypt rejects values outside this range.

Sample Output



```
faysal@DESKTOP-AOGE5FF:/mnt/c/Users/faysa/Dropbox/School/CSC220/Project1$ cat txt
Hi my name is faysal khatri
faysal@DESKTOP-AOGE5FF:/mnt/c/Users/faysa/Dropbox/School/CSC220/Project1$ cat txt | ./encrypt
⍰⍰h⍰ym⍰a⍰⍰m⍰si⍰a⍰⍰⍰⍰a⍰h⍰⍰⍰⍰faysal@DESKTOP-AOGE5FF:/mnt/c/Users/faysa/Dropbox/School/CSC220/Project1$ cat txt | ./encrypt | ./decrypt
HI MY NAME IS FAYSAL KHATRI
faysal@DESKTOP-AOGE5FF:/mnt/c/Users/faysa/Dropbox/School/CSC220/Project1$
```

encrypt.c

```
#include <stdio.h>
#include <string.h>
void encrypt();

char s[];
char key[] = "AZ";
int i;
int validInput = 0;

int main() {
  int sLen;

  scanf("%[^\n]s", s);
  validInput = 1;
  sLen = strlen(s);

  /*Convert to Upper Case*/
  for (i=0; i<sLen; i++) {
    s[i] = toupper(s[i]);
  }
  /*Only A-Z and spaces will be allowed*/
  for (i=0; i<strlen(s); i++) {
    if (!( (s[i] == ' ') || (s[i] >= 'A' && s[i] <='Z')) ) {
      validInput = 0;
      break;
    }
  }

  if (validInput) {
   encrypt();
  }
  else {
   printf("Invalid input detected. Only A-Z and spaces allowed.\n");
  }
  return 0;
}

void encrypt() {

  int keyLen = strlen(key);
  int sLen = strlen(s);
  unsigned char encrypted[sLen];

  /* encrypt using seed key */
  for (i=0; i<keyLen; i++) {
    encrypted[i] = (s[i] + key[i]);
  }

  /* encrypt using encrypted chars */
  for (i=keyLen; i<sLen; i++) {
    encrypted[i] = (s[i] + s[i - keyLen]);
  }

  printf("%s", encrypted);
}
```

decrypt.c

```
#include <stdio.h>
#include <string.h>
void decrypt();

unsigned char s[];
char key[] = "AZ";
int i;
int validInput = 0;

int main() {

  scanf("%[^\n]s", s);
  validInput = 1;

  for (i=0; i<strlen(s); i++) {
    if ( (s[i] < 64) || (s[i] > 180) ) { /* encrypted value range is 97 to 180 */
      validInput = 0;
    }
  }

  if (validInput) {
    decrypt();
  }
  else {
    printf ("Input cannot be decrypted.\n");
  }

  return 0;
}


void decrypt() {

  int keyLen = strlen(key);
  int sLen = strlen(s);
  char decrypted[sLen];

  /* decrypt using seed key */
  for (i=0; i<keyLen; i++) {
    decrypted[i] = (s[i] - key[i]);
  }

  /* decrypt using decrypted chars */
  for (i=keyLen; i<sLen; i++) {
    decrypted[i] = (s[i] - decrypted[i - keyLen]);
  }

 printf("%s\n", decrypted);
}
```