[R E P O R T]

정보통신공학전공

200301582

김성태

RC4 알고리즘 구현

1. 컴파일 방법

```
gcc -o RC4-3 RC4-3.c
```

2. 사용 방법

```
$ ./RC4-3 <암호화 혹은 복호화를 원하는 문자열> <문자열의 사이즈>
```

3. 소스 전문

```
#include <stdio.h>

#define MAXLEN 256

void swap(char *a, char *b);                    // switch function.
void initRC4(char *key, int size);              // RC4 keystream initialized.
void makeRC4(char *buffer, char *key, int size);    // do modulation

int main(int argc, char **argv)
{
        int stream_size, i;

        if(argc != 3) {
                printf("Usage : %s <data> <size>\n", argv[0]);
                return 1;
        }

        stream_size = atoi(argv[2]);

        char key[stream_size];

        initRC4(key, stream_size);
```

```
	makeRC4(argv[1], key, stream_size);


	for(i = 0; i < stream_size; i++)
		printf("%c", argv[1][i]);
	printf("\n");

	return 0;
}

/**********************
 * Name : swap          *
 * switching each others *
 **********************/
void swap(char *a, char *b)
{
	char temp;
	temp = *a;
	*b = *a;
	*a = temp;
}

/*******************************
 * Name : initRC4                *
 * initialized keystream         *
 *******************************/
void initRC4(char key[], int size)
{
	int i, j;
	char S[MAXLEN], K[MAXLEN];

	for(i = 0; i < MAXLEN; i++) {
		S[i] = i;
		K[i] = key[(i % size)];
	}
```

```
        for(i = 0; i < MAXLEN; i++) {
                j = (j + S[i] + K[i]) % MAXLEN;
                swap(&S[i], &S[j]);
        }
}

/*********************************
 * Name : makeRC4                *
 * Do modulation or demodulation  *
 *********************************/
void makeRC4(char *buffer, char *key, int size)
{
        int i, j, k, xor;

        for(i = 0; i < size; i++) {
                j = (j + 1) % MAXLEN;
                k = (j + key[i]) % MAXLEN;
                swap(&key[i], &key[j]);
                xor = (key[i] + key[j]) % MAXLEN;
                buffer[i] ^= key[xor];
        }
}
```

4. 실행 결과