

# Assignment #1

## CSE3029 암호학

### 2018-10-02

#### 1. 목표

AES 암호·복호화 알고리즘을 구현한다.

#### 2. 문제

제한사항에서 제시한 함수를 작성하여 AES 암호·복호화 알고리즘을 구현한다.

#### 3. 제한사항

과제의 함수명과 변수명은 반드시 다음과 동일하게 구현한다.

`void expandKey(BYTE *key, BYTE *roundKey)`

- 1) key: 키 스케줄링을 수행할 16바이트 키
- 2) roundKey: 키 스케줄링의 결과인 176바이트 라운드 키가 담길 공간

`BYTE* subBytes(BYTE *state, int mode)`

- 1) state: subBytes를 수행할 16바이트 입력 값, 수행 결과는 해당 배열에 바로 반영됨
- 2) mode: subBytes 수행 모드

`BYTE* shiftRows(BYTE *state, int mode)`

- 1) state: shiftRows를 수행할 16바이트 입력 값, 수행 결과는 해당 배열에 바로 반영됨
- 2) mode: shiftRows 수행 모드

`BYTE* mixColumns(BYTE *state, int mode)`

- 1) state: mixColumns를 수행할 16바이트 입력 값, 수행 결과는 해당 배열에 바로 반영됨
- 2) mode: mixColumns 수행 모드

`BYTE* addRoundKey(BYTE *state, BYTE *roundKey)`

- 1) state: addRoundKey를 수행할 16바이트 입력 값, 수행 결과는 해당 배열에 바로 반영됨
- 2) mode: addRoundKey 수행 모드

`void AES128(BYTE *input, BYTE *output, BYTE *key, int mode)`

mode가 ENC일 경우 평문을 암호화하고, DEC일 경우 암호문을 복호화하는 함수

[ENC 모드]

- 1) input: 평문 바이트 배열
- 2) output: 암호문이 담길 바이트 배열, 호출하는 사용자가 사전에 메모리 할당
- 3) key: 128비트 암호키 (16바이트)

[DEC 모드]

- 1) input: 암호문 바이트 배열
- 2) output: 평문이 담길 바이트 배열, 호출하는 사용자가 사전에 메모리 할당
- 3) key: 128비트 암호키 (16바이트)

## 4. 참고사항

필요한 경우 추가로 코드를 작성한다.

‘test\_AES128.c’는 암호·복호화 결과를 테스트 하기 위한 파일이므로 수정하거나 제출하지 않아도 된다.

## 5. 제출물

“학번\_AES128.zip” 형태로 제출한다.

과제의 소스 파일 (AES128.c, AES128.h 등) 앞에도 “학번\_”을 추가한다.  
마감일을 엄수하여 BlackBoard로 제출한다.

## 6. 마감기한

1) 마감일: 2018년 10월 14일 (일요일) 자정 전까지

2) 마감일을 넘겨 제출할 경우, 하루 단위를 총점의 20%씩 감점된다.

## 7. 참고자료

FIPS PUB 197, Advanced Encryption Standard (AES), <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>

## 8. 문의사항

[hseun@hanyang.ac.kr](mailto:hseun@hanyang.ac.kr), 메일 보낼 때 학번과 성명 표기해주세요

## - 결과 예제

```
- Plain Text :
6b c1 be e2 2e 40 9f 96 e9 3d 7e 11 73 93 17 2a
ae 2d 8a 57 1e 03 ac 9c 9e b7 6f ac 45 af 8e 51
30 c8 1c 46 a3 5c e4 11 e5 fb c1 19 1a 0a 52 ef
f6 9f 24 45 df 4f 9b 17 ad 2b 41 7b e6 6c 37 10
- Encrypted Plain Text :
76 49 ab ac 81 19 b2 46 ce e9 8e 9b 12 e9 19 7d
50 86 cb 9b 50 72 19 ee 95 db 11 3a 91 76 78 b2
73 be d6 b8 e3 c1 74 3b 71 16 e6 9e 22 22 95 16
3f f1 ca a1 68 1f ac 09 12 0e ca 30 75 86 e1 a7
=====
AES Encryption: SUCCESS!
=====
- Cipher Text :
76 49 ab ac 81 19 b2 46 ce e9 8e 9b 12 e9 19 7d
50 86 cb 9b 50 72 19 ee 95 db 11 3a 91 76 78 b2
73 be d6 b8 e3 c1 74 3b 71 16 e6 9e 22 22 95 16
3f f1 ca a1 68 1f ac 09 12 0e ca 30 75 86 e1 a7
- Decrypted Cipher Text :
6b c1 be e2 2e 40 9f 96 e9 3d 7e 11 73 93 17 2a
ae 2d 8a 57 1e 03 ac 9c 9e b7 6f ac 45 af 8e 51
30 c8 1c 46 a3 5c e4 11 e5 fb c1 19 1a 0a 52 ef
f6 9f 24 45 df 4f 9b 17 ad 2b 41 7b e6 6c 37 10
=====
AES Decryption: SUCCESS!
=====
```