# 어셈블리프로그램 설계 및 실습 - term project

담당교수 : 이준환 (컴퓨터공학과)

2020년 2학기

"꿈을 가지십시오. 그리고 정열적이고 명예롭게 이루십시오."

이번 학기 설계 과제는 임의의 정수 데이터로 이루어진 NxN(10<=N<=20) 정방행렬에 대하여 컨볼루션 연산과 맥스풀링 연산을 수행하는 어셈블리 코드를 작성하는 것이다. 과제의 자세한 specification을 충분히 이해하여 과제를 수행하도록 한다.

## 1. System Overview

프로젝트의 전체 동작은 임의의 정수 데이터로 이루어진 NxN 정방행렬에 대한 컨볼루션 연산과 맥스풀링 연산을 수행함에 있어, 성능이 가장 좋은 코드를 구현하는 것이다. 이를 위해 필요한 알고리즘의 사용에는 제한이 없다. 성능의 기준은 code size와 state<sup>2</sup>을 곱한 값으로, 결과 값이 작을수록 성능이 좋다고 정의한다.

#### 1.1 Convolution

컨볼루션 연산은 입력 데이터와 커널의 곱으로 이루어진다. 컨볼루션 연산의 결과는 커널을 일정한 간격(Stride)으로 이동해가며 입력 데이터와 곱하여 얻을 수 있다. 커널의 크기를 KxK라고 할 때, (i,j) 위치에서의 출력 데이터 Output의 값은 수식 (1)과 같이 구할수 있다.

$$Output[i,j] = \sum_{x=0}^{k-1} \sum_{y=0}^{k-1} Input[i+x,j+y] Kernel[x,y]$$
 (1)

컨볼루션 연산을 수행하는데 있어, 출력 데이터의 크기를 입력 데이터의 크기와 맞추기 위하여 입력 데이터에 대하여 패딩(padding)이 수행될 수 있다. 하지만 본 프로젝트에서 는 패딩을 수행하지 않는다.

## 1.2 Max pooling

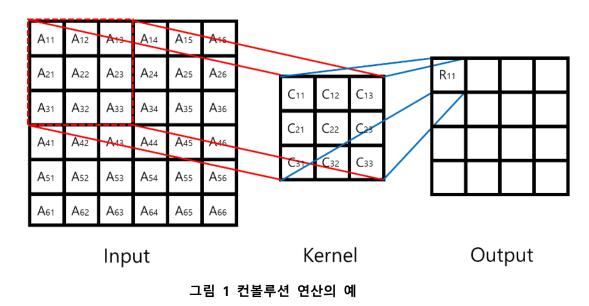
맥스풀링 연산은 연산을 수행하는 영역에서 최대값을 찾는다. 맥스풀링 또한 컨볼루션과 유사하게 맥스풀링 영역을 일정한 간격(Stride)으로 이동해가며 연산을 수행한다. 이때, 컨볼루션과 달리 맥스풀링의 stride는 풀링 윈도우의 너비와 같다. 풀링 윈도우의 크기를 KxK라고 할 때, (i,j) 위치에서의 출력 데이터 Output의 값은 수식 (2)과 같이 구할 수 있다.

$$Output[i,j] = \max_{x,y=0}^{k-1} Input(i+x,j+y)$$
 (2)

위에서  $\max(x)$  에는 x 값 중 최대값을 return한다.

담당교수 : 이준환 (컴퓨터공학과) 2020년 2학기

다음 그림 1과 2는 6x6 정방행렬에 대하여 3x3 컨볼루션 연산(stride 1)과 2x2 맥스풀링 연산 과정의 일부분에 대한 예시를 보여준다.



컨볼루션 연산은 입력된 6x6 정방행렬에 대하여 3x3 커널을 1의 간격(stride)으로 이동해가며 수행한다.

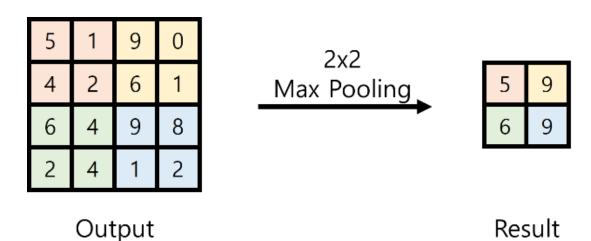


그림 2 맥스풀링 연산의 예

맥스풀링 연산은 컨볼루션 연산의 결과인 Output에 대하여 수행된다. Output의 결과가 그림2와 같을 때, 수식 (2)과 같이 맥스풀링 연산을 수행하면 그림과 같은 결과를 얻는다.

- <u>구체적인 설계 사양은 업데이트 될 수 있습니다. 업데이트 될 때마다 종합정보서비스</u> 의 해당 과목 공지사항에 공지할 테니 확인하시기 바랍니다.

# 2. Specification

- 2.1 Project specification
- Label: Project\_data
  - ✓ DCD 명령어를 이용하여 데이터가 선언되어 있음
  - ✓ 컨볼루션 커널의 크기, stride, 각 원소의 값이 저장되어 있음
  - ✓ 맥스풀링 윈도우의 크기가 저장되어 있음
  - ✓ 컨볼루션 커널의 크기, stride, 맥스풀링 윈도우 크기는 연산 시 나머지가 없는 경 우만 발생한다고 가정

담당교수: 이준환(컴퓨터공학과)

2020년 2학기

✓ 임의의 데이터는 -2000 ~ 2000 사이의 정수 값을 갖음, 단 커널의 경우 커널의 원소 중 30%(내림)는 0 값을 갖음

Ex) 4x4 커널의 경우 30%의 원소의 수는 4개

✓ 데이터 예시

; Convolution size DCD 3 : Stride size DCD 1 ; Max Pool size DCD 2 ; Matrix size DCD 6 DCD 2\_00101010001010000101010001001 ; Matrix element A<sub>11</sub> DCD 2\_00101010001010000101010001001 ; Matrix element A<sub>12</sub> DCD 2\_00101010001010000101010001001 ; Matrix element A<sub>13</sub> DCD 2\_00101010001010000101010001001 ; Matrix element A<sub>14</sub> DCD 2\_00101010001010000101010001001 ; Matrix element A<sub>15</sub> DCD 2\_00101010001010000101010001001 ; Matrix element A<sub>16</sub> DCD 2 00101010001010000101010001001 ; Matrix element A66

#### Label: Result data

- ✓ Matrix\_data의 원소들을 이용하여 컨볼루션과 맥스풀링 연산을 수행한 결과가 저 장되는 데이터 공간의 시작 주소 값
- ✓ 데이터를 저장할 때에는 1 word 단위로 저장하며, 0x60000000 번지로 통일
- ✓ 결과 행렬의 데이터는 Project\_data와 동일한 순서로 저장
  Ex) 2x2 행렬의 경우, R<sub>11</sub>, R<sub>12</sub>, R<sub>21</sub>, R<sub>22</sub> 순서로 저장
- 주어진 데이터를 정렬하는 과정에서 Matrix\_data와 Result\_data를 제외한 임시적인 데이터 공간의 사용은 자유
- 임의의 데이터에 대한 예제들은 제공될 예정
- MUL 명령어 사용 금지
- 프로그램은 주석을 포함하여 다음과 같이 종료
  - ✓ MOV pc, #0 ;Program end

담당교수 : 이준환 (컴퓨터공학과) 2020년 2학기

## 3. Important dates

- Issue date: 11월 17일
- 제안서 제출
  - ✓ 11월 23일 23시 59분까지 klas 과제 제출에 제출
- 최종 결과 및 보고서 제출
  - ✓ 11월 30일 23시 59분까지 종합정보서비스 과제 제출에 제출
  - ✓ ini 파일을 포함(ini 관련 문제로 정상 동작 안될 경우 0점 처리)
- Project 결과 발표: 12월 1, 3일 수업시간
- 발표 희망자는 박준택 조교에게 메일 보낼 것(juntaek@kw.ac.kr). (선착순)

# 4. Report outline

- 4.1 제안서
- 보고서: 최소한 다음의 내용들이 포함되어야 하며 그 외의 것을 추가하는 것은 자유 ✓ 과제 제목, 과제 목표, 일정, 각 function 별 알고리즘, 예상되는 문제점, 검증 전략 등

## 4.2 결과 보고서

- 발표: PoewrPonit로 발표 희망자에 한해 10분 분량으로 작성
- 보고서: 최소한 다음의 내용들이 포함되어야 하며 그 외의 것을 추가하는 것은 자유
  - Introduction
    일정 및 계획을 포함할 것
  - 2. Project Specification
  - 3. Algorithm
  - 4. Performance & Result
  - 5. Conclusion과제 완료 후 기대되는 학습 효과를 포함할 것
- 보고서의 가장 ideal한 양식은 논문의 형태 입니다.

## 5. 프로젝트 점수 분배

- 제안서: 20%

- 코드: 30%

- Performance: 20%

- 결과보고서: 30%