

TeX 사용에서 주의할 점

김 영 욱
고려대학교

Abstract

이 글은 일본의 교수님 한 분이 TeX 사용에서 주의할 점을 정리해 놓은 글을 읽고 우리에게 필요한 것을 적은 것이다. 많은 내용은 이 분이 적은 글과 겹친다. 이 글은 양성덕교수님과 함께 기획하고 있는 L^AT_EX 사용법 안내서의 일부로 기획된 것이다. 양교수님이 시간이 나면 기획하고 있는 이 문서를 만들어 나갈 것이다.

이 글에서 지적된 몇 가지 문제는 최근에 개발된 `kotex.sty` 패키지에서는 해결되었다. 특히 영문이 끝나고 붙는 우리말 조사 사이의 간격 등은 이 패키지의 `finemath` option을 사용하면 예쁘게 출력된다.

지금까지 L^AT_EX을 사용하여 한글 문서를 만드는 법을 알아보았다면, 여기서는 한글 문서를 만들 때에 문서를 조금 더 보기 좋게 하는 방법과 글자를 배열하는 규칙에 대하여 말하고자 한다.

글자를 조판하는데는 오랜 기간에 걸쳐 만들어진 관습이 몇 가지 있다. 이들은 일견하여 큰 문제가 되지 않을 법한 자잘한 것들이어서 많은 사람들이 이를 무시하고 사용하는 경우가 많다. 그러나 문서를 만들어본 경험이 늘어나면서 이러한 자그만 부분들이 서로 달라서, 여러 군데서 작업한 문서가 모이면 통일성이 떨어지는 점을 발견하게 된다. 특히 조판에 엄격한 원칙을 지키고 있는 서양의 논문집 등에 투고하려고 할 때에는 그들이 요구하는 내용의 의미 조차 파악되지 않는 경우도 있는 것은 처음부터 문서 작성에서 이러한 점들을 간과하는 데 익숙해진 결과라고 할 것이다.

아래에 언급하는 몇 가지 조언으로 모든 문제가 해결되는 것은 아니다. 그러나 이것들 만큼은 적어도 지키는 것이 글틀로 한글 수학문서를 만들 때에 통일되고 예쁜 문서를 만드는 한 가지 방법이 될 것이라고 생각한다.

1 여러 가지 기호와 간격

L^AT_EX를 사용하면서 가장 먼저 느끼게 되는 것은 한글을 사용하는 것을 고려하지 않고 만들었다는 것이다. 여러 경우에 적절치 못한 간격으로 눈에 거슬리는 문서가 만들어지게 된다.

이 가운데 가장 문제가 되면서도 쉽게 고치기 힘든 것이 우리말 가운데 영어 문자가 나왔을 때 이 뒤에 따라 붙는 조사와의 사이 간격이다. 영어문자와 조사 사이를 한 간을 떼는 것은 너무 간격이 크며 그렇다고 떼지 않으면 너무 붙어 보인다는 느낌을 지울 수 없다. 그렇다고 매 번 ‘\,’와 같은 기호를 붙이는 것은 번거로울뿐 아니라 마침 이 자리에서 줄을 바꾸게 될 경우에는 이 간격이 줄 끝에 붙어 보기 싫은 문서가 된다. 이 문제는 점차로 해결되어야 할 것이다.

그러나 다음에 이야기하는 것들은 조금만 주의하면 쉽게 쓸 수 있는 것들이다.

1.1 구두점

글을 쓸 때 가장 주의해야 하는 것이 구두점이다. 구두점으로는 마침표(period), 쉼표(comma), 콜론, 세미콜론 등이 있으며 그 사용법은 경우에 잘 맞추어 사용하여야 한다. 이 기호들은 그 앞에 오는 글자와의 사이에 빈 간을 두면 안 된다. 괄호를 열고 닫을 때도 괄호와 그 안의 내용 사이에 빈 간을 두면 안 된다. 예를 들면,

...을 보면 (Cf. [8]), ... 는 틀리고, ...을 보면(Cf. [8]), ...

가 맞으며,

...다음과 같다: 는 틀리고, ...다음과 같다:

라고 하여야 한다.

한편, 이름의 머리글자만을 따서 줄여 쓰는 경우(이니셜)에는 조금 주의하여야 한다. 우선 이니셜에 쓰이는 마침표 다음에는 빈 간을 두어야 한다.

R.E.Greene 이나 S.D. Yang

은 옳지 않다. (이에 대하여 e.g., a.e.나 i.e. 등과 같이 관습이 된 경우는 예외로 한다.) 따라서 이름을 쓰는 경우는 R. E. Greene 이나 S. D. Yang과 같이 써야 한다.

여기서 한 가지 더 주의하여 둘 것이 있다. 위와 같이 줄여쓰기 위하여 \TeX 에서 ‘J. Math. Soc. Korea’라고 입력하게 되면

J. Math. Soc. Korea

과 같은 표현이 된다. 잘 살펴 보면 ‘Math.’ 뒤의 간격이 보통 띄어쓰기 간격보다 넓다는 것을 알 수 있다. 이것은 \TeX 이 소문자 뒤의 마침표와 띄어쓰기가 연달아 나오는 경우에는 default로 한 문장이 끝난 것으로 보기 때문에 이 띄어쓰기를 한 문장 안에서 단어 사이의 띄어쓰기 간격보다 큰 문장 사이의 띄어쓰기 간격에 맞추었기 때문이다.(대문자 뒤의 마침표는 제대로 줄여쓰기를 한 것이라고 알아 듣는다.) 간격을 제대로 맞추어 쓰려면 간격이 보통의

띄어쓰기 간격임을 가르쳐 주어야 한다. 즉, ‘J. Math.\ Soc.\ Korea’라고 입력하면

J. Math. Soc. Korea

J. Math. Soc. Korea

의 첫째 줄과 같이 바른 표현이 된다. 둘째 줄의 틀린 표현과 비교하여 보자.

이 밖에 주의할 점은 수식에서 $\ldots(\dots)$ 와 $\cdots(\dots)$ 를 구별하여 쓰는 것이다. 이는 다음 예에서 보면 알 수 있다:

$$x_1, x_2, \dots, x_n, \quad \prod x_i = x_1 x_2 \cdots x_n, \quad \sum x_i = x_1 + x_2 + \cdots + x_n.$$

1.2 수식에서의 글자체

수식 안에서 사용되는 기호 가운데는 어떤 단어를 줄여서 사용한 경우가 있다. 주로 함수나 작용소의 이름이 이런 경우에 해당되며 많은 함수가 이미 \TeX 안에 `math operator` 기호로 정의되어 있다. 전체 리스트는 부록에서 찾아볼 수 있지만 몇 가지만 예로 들면 `log`, `sin`, `max` 등이다. 이러한 것들은 수식 안에서 이탤릭체를 쓰지 않음으로써 변수들과 구별한다. 또 이 작용소는 뒤에 따라오는 변수와 사이에 간격을 두고 있다. 따라서

`span`{ v_1, v_2, \dots, v_n }, `Hom`(R, S), `log` x 등은 틀리고,

`span`{ v_1, v_2, \dots, v_n }, `Hom`(R, S), `log` x

가 맞는 표현이다. 수식에서

`log` x , `log` x , `sin` x , `Diff`(M)

등과 같이 쓰면 안된다. 여기서 \LaTeX 안에는 `\span`이나 `\Hom`과 같은 명령어가 없다. 이를 제대로 나타내려면 다음과 같이 한다:

`\mathop{\mathrm{span}}\nolimits` $\{v_1, v_2, \dots, v_n\}$.

AMS에서 만든 여러 명령어 가운데는 이를 간단하게 하여 주는 ‘`\operatorname`’이라는 명령어가 있다. 위의 표현은

`\operatorname{span}` $\{v_1, v_2, \dots, v_n\}$

라고 간단히 나타낼 수 있다.

작용소는 아니지만 항상 로마체로 나타내야 하는 기호들에는 다음과 같은 것들이 있다:

`ess`, `Id`, `id`, `ns`, `pt`, `red`, `reg`, `top`, `ur`,

`a.e.`, `i.e.`, `e.g.`, `I`, `II`, `III`, \dots , `i`, `ii`, `iii`, \dots

1.3 theorem 환경 안에서

1.3.1 영어를 쓸 때

theorem 환경에서 글을 쓰면 글자가 이탤릭체로 나타난다. 이러한 가운데 보면 이탤릭체로 쓰면 어색한 부분이 눈에 띈다. 괄호와 숫자 같은 것 들이다. 즉,

... (Cf. Mumford[3,§2]), the condition(R)

은 옳은 표현이 아니다. 이는

... (Cf. Mumford[3,§2]), the condition(R)

와 같이 나타내어야 한다. theorem 환경 안에서 이렇게 나타내려면

\ldots\textup{({}Cf. Mumford\textup{[3,\S2])}},
the condition\textup{(R)}

과 같이 쓰면 된다.

1.3.2 한글을 쓸 때

한편 정리 환경 안에서 우리글이 이탤릭체로 누워 있는 것은 보기에 좋지 않다. 다른 글꼴로 바꾸는 경우에도 그렇지만 그대로 명조체를 쓰는 경우에는 \upshape 명령을 써서 바른 글씨체를 쓰는 것이 옳아 보인다. 영어의 이탤릭체에 해당하는 것은 우리글자체에서는 필기체가 가까워 보인다. 다음을 비교하여 보자.

정리 1 사영기하에서는 참인 명제의 쌍대명제도 항상 참이다.

정리 2 사영기하에서는 참인 명제의 쌍대명제도 항상 참이다.

정리 3 사영기하에서는 참인 명제의 쌍대명제도 항상 참이다.

1.4 특수한 기호들

1.4.1 하이픈과 대쉬

영어에서 두 단어를 연결하여 한 단어로 만들때 사용하는 기호를 ‘하이픈(hyphen)’이라고 부른다. 이 기호를 우리나라에서는 보편적으로 ‘대쉬’ 또는 ‘다시’라고 부르는 경향이 있다. 그러나 대쉬(dash)는 하이픈보다는 조금 긴 기호이고 구별되어 사용된다. TEX에는 이러한 기호가 세 가지가 있다. 첫번째 것은 hyphen이고 semi-riemannian과 같은 경우가 그것이다. 이것은

semi-riemannian

와 같이 입력하면 된다.

이 밖에 대쉬로는 en대쉬와 em대쉬가 있다. en대쉬는 로마자 n자의 폭과 같은 길이의 대쉬이며 주로 쪽수를 연결하는데 쓰인다. 즉 ‘p.\ 35--48’이라고 입력하면 ‘p. 35-48’과 같이 나타내어진다. 우리는 ‘35~48쪽’이라고 쓰는 것을 더 좋아하지만.

한편 로마자 m자의 폭과 같은 길이의 em대쉬는 문장—삽입되는 구 등—을 연결하거나 분리하는 데 쓰인다. 기호로는 ‘---’를 사용한다.

주의하여야 할 점은 math mode(수식) 안에서는 ‘-’기호를 쓰면 자동적으로 마이너스 기호(‘-’)가 된다는 것이다. 즉, ‘ \mathbb{R} -module’이라 입력하여 ‘ \mathbb{R} -module’과 같이 써야 하며 ‘ \mathbb{R} -\$module’이라 잘못 입력하여 ‘ \mathbb{R} -module’이라고 쓰면 안 된다.

1.4.2 내적

수식에서 내적을 나타낼 때 ‘ $\langle v, w \rangle$ ’라고 쓰는 경우가 흔하다. 그러나 이렇게 쓰면 ‘ $\langle v, w \rangle$ ’와 같이 나타나게 된다. 그러나 TeX은 내적과 같은 기호를 위해서 ‘ $\langle v, w \rangle$ ’이라는 기호를 만들어 두었다. 즉, ‘ $\langle v, w \rangle$ ’이라고 입력하면 ‘ $\langle v, w \rangle$ ’로 출력되며 이것이 올바른 내적 기호이다.

수식 안에서 ‘ \langle ’ 기호를 사용하게 될 때 주의할 사항 하나는 이 기호가 binary operator로서 인식되고 있다는 것이다. 다시 말하면 이 기호는 그 왼쪽, 오른쪽의 기호 또는 문자들과의 사이 간격이 넓게 출력된다는 것이다. 다음과 같다:

$2 < 3$ 이라고 입력하면 $2 < 3$ 이라고 출력되고,

$2 \langle 3$ 이라고 입력하면 $2 \langle 3$

이라고 출력된다. 굳이 내적에 부등호 기호를 사용해야 하겠다면, $\langle v, w \rangle$ 보다는

$\langle v, w \rangle$ 라고 입력하여

$\langle v, w \rangle$ 라고 나타내는 것이 좋아 보인다.

1.4.3 공집합 기호

공집합을 나타내는데 자주 쓰이는 ϕ 는 그리스 문자 파이(phi)의 소문자이며 이는 공집합을 나타내는 기호가 아니다. 공집합을 나타내는 기호는 \emptyset 을 변형한 것으로 ‘ \emptyset ’를 사용하여 \emptyset 로 나타내어야 한다. 이 기호 말고도 ‘ \varnothing ’이라는 명령어가 있어서 \emptyset 을 출력하게 해 준다.

1.4.4 집합의 뺄셈 기호

집합의 뺄셈 기호에 ‘ \backslash ’를 사용하면 안 된다. backslash는 이항연산 부호가 아니여서 뺄셈을 하는 두 집합기호와의 사이가 너무 붙어버린다. 즉,

‘ $\backslash X \backslash Y$ ’라고 입력하면 $X \setminus Y$ 를 얻게 된다. 집합의 뺄셈 기호를 위하여는 ‘ $\backslash setminus$ ’라는 명령어가 있으며, ‘ $\backslash X \setminus Y$ ’라고 입력하면 $X \setminus Y$ 를 출력한다. 혹 이 뺄셈 기호가 너무 기울어져 있어서 일반적으로 사용하는 기호에 가까운 것을 원하면 AMSTeX의 기호인 ‘ $\backslash smallsetminus$ ’를 사용하기 바란다. 이 기호를 출력하면 $X \setminus Y$ 와 같다.

1.4.5 여러 가지 작용소

합집합과 교집합을 나타낼 때는 ‘ $\backslash cup$ ’ 또는 ‘ $\backslash cap$ ’을 사용하지만, index를 사용하여야 할 경우에는 ‘ $\backslash bigcup$, $\backslash bigcap$ ’ 등을 사용하여 $\bigcup_{i \in I} X_i$ 등과 같이 나타내어야 한다. 외적을 비롯한 다른 여러 작용소에도 **big** 버전이 있으며 이는 $\wedge, \bigwedge, \vee, \bigvee, \oplus, \bigoplus, \otimes, \bigotimes$ 등으로 차이가 난다. 외적을 나타내는데 ‘ $\backslash Lambda$ ’(Λ)를 사용하면 안 된다.

1.5 우리말로 된 수학 문서

여러분은 TeX을 사용하여 보면서 우리가 고등학교 때까지 흔히 접하던 책들과는 조금 다르다는 느낌을 받을 것이다. 가장 눈에 띄는 것이 기호로 쓰이는 글자들이 이탤릭체로 쓰이고 있다는 것이다. 수식은 일반적으로 이탤릭체로 쓰고 있었기 때문에 크게 문제되어 보이지 않는다. 그러나 ‘점 A’와 같은 것은 ‘점 A’로 사용하여 왔던 습관 때문에 오히려 어색하게 보인다.

우선 생각하여 볼 것은 왜 수식을 이탤릭체로 쓰게 되었는가 하는 것이다. 그리고 이에 대한 답은 영어로 된 문장의 틈에서 수식이 눈에 들어나게 하려니까 글씨체를 바꾸어야 할 필요에서 이탤릭체를 쓰게 되었다는 것이 정설이다. 이제 우리말로 된 수학문서를 작성함에 있어서 생각하여 보면 우리글 틈에서 로마자를 사용한 수식을 구별하는 것을 특별한 서체를 필요로 하지 않는다. 어디서나 외국 글자들은 잘 두드러져 보인다. 따라서 특별히 이탤릭체를 쓸 필요는 없는 것이다. 그래서 중학교 교과서에서 점을 나타낼 때 ‘점 A’라고 표기했던 것이다. 그러나 TeX을 사용하며 보면 수식을 사용하는 중에 매번 upright형의 로마자를 쓰는 것은 불편하다. 어쩔 수 없이 이탤릭체를 사용하게 될지도 모른다.

이러한 생각을 하다 보면 영어책과 우리말 책을 비교하여 보게 된다. 많은 우리말 책의 조판이 소잡한 점은 있지만 우리말 책을 들여다 보면 그 책의 내용의 구조가 확연하게 눈에 들어옴을 알 수 있다. 이는 우리가 우리글에 익숙해서가 아니다. 한 글자 한 글자 읽지 않고 한 쪽의 내용을 한 눈에 보아도 어디가 수식이고 어디가 내용인지 바로 눈에 보인다. 영어로 쓰인 책을 볼 때에는 전혀 다르다. 이는 서양사람에게도 마찬가지일 것임에 틀림이 없으며, 이는 우리가 서양사람에 대하여 갖는 단 하나의 유리한 조건일 지도 모른다. 우리말로 수학 문서를 쓰는 것은 쉽지 않은 작업이지만 그러한 일을 할만한 가치가 있다고 생각되지 않는가?