韓國數學敎育學會誌 시리즈 A <數學敎育>
1996. 6. 제 35권, 제 1호, 15-23.
Journal of the Korea Society of Mathematical Education
Series A: Jun. 1996, Vol. 35, No. 1, 15-23.

수학교육에서의 컴퓨터의 이용

Mathematica를 이용한 미적분 지도를 중심으로[1)]

황 일 (건국대)

I. 서 론

미적분학은 고등수학의 연구를 위해서 중요한 비중을 차지하고 있으며 동시에 응용면이 많기 때문에 대학수학의 입문과정으로 자연·이공계열 학생들에게 필수적인 교과내용이 되어 오고 있다. 전통적으로 미적분 강의는 미적분의 기본적인 개념들을 이해하고, 문장이나 기호로 주어진 미적분 문제를 풀도록 하는데 초점이 맞추어져 있다. 따라서 주어진 문제들은 기호로 바꾸었을 때 가능한 한 간결한 계산으로 풀 수 있는 것들에 한정되어 있으며 강의와 과제들은 알고리즘의 습득에 큰 비중을 두고 있었다.

그러나 계산기나 컴퓨터의 발달되고 널리 보급되었기 때문에 미적분학의 학습에서 학생들이 풀어야 할 문제의 복잡성이나 수량들을 한정할 필요가 없게 되었다. 사실 기호를 사용하고 문제를 푸는 것과 그 의미를 이해한다는 것은 별개의 문제이다. 함수, 극한, 연속, 미분가능성, 변화율, 등과 같은 미적분의 중심적인 개념들은 그래프와 수량적인 접근에 의해 그 본질이 더 분명하게 파악될 수 있다. 이러한 접근은 직관력을 증진시키며 미적분을 처음 배우는 학생들에게 어려운 개념들을 예시하고 강화해 줌으로써 이해를 도울 수 있다.

Mathematica는 수학의 교수·학습을 목적으로 개발된 프로그램의 하나이다. Mathematica는 운용 방법에 따라 수학적인 사실이나 원리를 실연(demo)하거나 탐구를 위한 도구로 사용될 수도 있다.

본 연구는 대학에서 자연계열 1학년 학생들이 필수적으로 수강하고 있는 미적분강의를 단순한 알고리즘의 습득이 아닌 수학적인 탐구의 기회로 활용하기 위한 방법의 일환으로, Mathematica를 이용한 수업모형을 개발하고, 그 효용성을 탐색해 보고자 한다.

II. 이론적인 배경

1. 컴퓨터와 미적분학 강의

컴퓨터의 발달로 복잡한 계산을 보다 신속 정확하게 하고 방대한 양의 자료를 저장하고 신속하게 이를 처리할 수 있게 되었다. 이러한 컴퓨터 과학과 컴퓨터의 발달은 수학에 있어서 수치적 방법 같은 새로운 기법이 가능하게 되었으며, 수학적인 활동 자체를 보다 실험적으로 변화시켰다. 컴퓨터는 수학적 대상과 개념을 모의실험하고 시각화하여 줌으로써, 증명에 앞서 추측을 하고 종합할 수 있게 해 주기 때문이다.

컴퓨터와 컴퓨터과학의 영향으로 학교 수학에서 계산기와 컴퓨터를 적극적으로 도입하려는 시도가 이루어 지고 있는데 이는 세계적으로 공통된 추세이다.

최근에 세계 여러 나라에서 사용하고 있는 미적분학의 교재들은 해석기하를 포함시키며, 컴퓨터 소프트웨어를 내용에 통합시켜 사용할 수 있도록 저술되어 있다. 예를 들면, Thomas와 Finney(황일, 1995)의 Calculus and analytic

---

1) 1995년도 건국대학교 학술진흥처 연구비 지원에 의한 논문임.

geometry에서는 Analyzer, Insight, Mathematica, Derive, MathCad 등의 소프트웨어를 사용한 활동이나 BASIC 등의 언어를 이용한 프로그래밍의 일부를 옵션으로 내용에 포함시키고 있다. 수학의 교수·학습에 컴퓨터 소프트웨어를 사용한다는 것은 학생들이 다루는 문제들이 지금까지의 양상과 확실하게 구별이 됨을 의미한다. 즉, 기호로 접근하며 계산을 최소화하고, 대수식의 경우에는 쉽게 인수분해되는 것들을 다룬다든지, 각의 계산이 관련된 문제에는 15°, 30°, 45° 등 15의 배수만을 취급하는 등의 제한이 필요하지 않게 된 것이다. 최근의 미적분학의 교재에는 Newton방법 또는 Picard 방법들을 소개하고 실제로 수치적 방법의 적용을 실연함으로써 컴퓨터를 미적분학의 학습의 장에 적극적으로 끌어 들이고 있음을 알 수 있다.

2. 컴퓨터의 사용에 관한 교육학적인 고려

컴퓨터는 교사를 돕는 교수의 방법으로 생각될 수 있다. 교수에서 컴퓨터를 사용하는 방법은 여러 가지이다. 컴퓨터는 모니터의 화면을 확대하여 별도의 스크린에 투사할 수 있는 보조 장치들을 이용함으로써 칠판대용으로 사용될 수 있다. 교실에 컴퓨터 한 대를 교사가 조작하여 보여주거나 예를 제시하거나 그림을 그릴 수 있다. 또 교실에 개인용 컴퓨터(PC)가 있어서 한 두명 혹은 몇 명의 그룹이 되어 프로그래밍을 하거나 팩키지를 사용할 수 도 있다. 여러 대의 컴퓨터를 네트워크로 연결할 수도 있다. 또 컴퓨터실에서 "실질적인 작업"에 컴퓨터를 이용할 수도 있다.

질문지와 시험을 통해 학습자로 하여금 정해진 과정을 통과하게 하는 "개인교습"형태의 CAI (Computer Aided Instruction: 컴퓨터 보조학습)가 전통적으로 컴퓨터가 학교수학에서 활용되는 방법이다. 그러나 현재는 프로그래밍을 통한 학습, 그래픽 패키지나 수학적인 개념 등을 보여 주거나, 모의실험하는 소프트웨어들, 수학적인 목적을 위해서 스프레드 쉬트(spread sheet), 사용자가 대상들을 조정할 수 있는 소우주(micro world) 등의 수학의 교수를 보조하기 위한 소프트웨어로 개발되어 활용되고 있다. 컴퓨터는 학습자와의 관계에서 활용수준에 따라 CAI에서와 같이 교사(tutor)로, 사용자가 언어를 이용하여 프로그래밍을 함으로써 컴퓨터가 작업 내용을 배우는 학습자(tutee)로, 또는 도구(tool) 수준에서 사용될 수 있다(Taylor, 1980). 최근의 많은 소프트웨어들은 사용자가 탐색할 수 있도록 학습 보조수단으로 사용하고 있는 추세이다. 상황이 즉각 학생들의 학습환경을 통제하는 탐색적인 방법으로 사용될 때, 적극적인 학습이 장려되며 학생들은 스스로 생각하고 발견하기 시작하게 된다.

컴퓨터의 사용은 언제나 학습에 도움이 되는가? 이는 교사나 소프트웨어 제작자들이 반드시 염두에 두어야 할 일이다. 예를 들어 그래픽 소프트웨어를 사용하면 기하학적인 도형들을 정확하게 빠른 시간 내에 얼마든지 그릴 수 있다. 그러나 그것이 학습을 돕는다는 것과는 별개의 문제이다. 소프트웨어의 활용이 학습을 향상시킨다는 보장은 없으며 여기에서 교수학적인 전략이 매우 중대한 역할을 한다. 문제는 컴퓨터의 사용이 학생들의 지식과, 이 지식이 구조화되는 방식, 이 지식을 활용하는 학생들의 방식에 어떤 영향을 미치는가 하는 것이다.

3. Mathematica의 기능과 특징

Mathematica는 C 언어로 쓰여진 프로그램으로 Wolfram(1991)의 책의 부제("A system for doing mathematics by computer")에서 표방하고 있는 바와 같이 수학을 컴퓨터로 하기 위한 목적으로 개발되었으며 현재는 공학을 위한 목적으로 많이 활용되고 있다.

Mathematica는 BASIC과 같은 수 계산의 기능, Maple과 같은 대수체계로서의 기능, 그래픽

기능, LISP같은 List 조작기능, C나 Pascal과 같은 프로그래밍언어 기능을 갖고 있다(Wolfram, 1991, xvii).

(1) 수 계산의 기능

계산기와 같은 산술계산이 가능하며, 함수 N을 이용하면 근사값 계산이 가능하여 수학적인 대상에 대하여 수치적인 방법 (Numerical method)을 적용할 수 있다. 계산기와는 달리 실수 뿐 아니라 복소수를 대상으로 하는 계산도 가능하며, 영역별로 함수 package를 가지고 있어서, 미적분 뿐 아니라, Zeta 함수, Bessel 함수 등 표준 수학함수 모두를 계산 할 수 있다.

(2) 대수체계로서의 기능

일반적으로 수학책에서와 달리 컴퓨터 프로그램에서만 통용되는 수식의 표현이 있는데, 예를 들면 $x^3-\frac{1}{3}x+\frac{1}{x}$ 을 x^3-(1/3)x+1/x 로 나타내는 것 등이다. Mathematica는 이렇게 후자와 같이 입력한 수식을 $x^3-\frac{1}{3}x+\frac{1}{x}$ 와 같이 수학책에서 볼 수 있는 표준적인 형태로 나타내 준다.

Mathematica는 수의 계산 뿐 아니라 문자식도 다룰 수 있다. 다항식의 전개나 인수분해, 대입에 의한 식의 계산, 식의 정리, 다항식에서 주어진 문자의 계수를 찾아 내기도 하며, 다항방정식이나 미분방정식의 해를 내장된 함수를 이용하여 구해 내기도 한다.

또, x가 0에 가까워 질 때 임의의 항까지, 예를 들면 x^3의 항까지 $(1+x)^n$ 에 근사하는 멱급수

$$1+nx+\frac{(-1+n)\,nx^2}{2}+\frac{(-2+n)(-1+n)\,nx^3}{6}+O[\ x]$$

를 계산해 낼 수도 있다.

(3) 그래픽과 Sound기능

Mathematica는 극좌표나 직교좌표, 또는 매개변수로 표현된 식의 그래프를 한꺼번에 같은 좌표평면에 그릴 수 있으며, 독립변수가 두 개인 경우에 3차원의 그래프를 그릴 수도 있으며 관점이나 음영을 옵션으로 달리 조정함으로써 여러 각도에서 바라 본 그래프를 얻을 수 있다. 여러 개의 일련의 그래프를 동화상으로 보여 줄 수도 있다. 그밖에 등위곡선이나 Density plot도 그릴 수 있는데 기본적인 기능을 제외하고 'graphic' 패키지를 실어서 작동시키게 된다.

또 f(x)=sin 10π t와 같은 파형함수(wave form)를 입력했을 때 그 함수가 나타내는 파(wave)의 주파수와 파고에 해당하는 음을 나타내는 Sound기능이 있다.

(4) List 조작기능

Mathematica는 수값의 list를 출력하기도, 길이나 차원 등 List의 구조를 규명하기도 한다. 또 어떤 원소가 list에 포함되어 있는 지의 여부 등을 판단하기도 하며, list의 특정 번째의 원소를 찾아 내기도 한다. 여러 개의 list들을 통합하거나 재배열할 수 있으며 원소들을 분할하기도 한다. 벡터나 행렬을 list로 나타내어 여러 가지 연산을 할 수 있으며, list에서 배열순서를 무시하면 집합으로 취급할 수도 있다.

(5) C나 Pascal 같은 프로그래밍언어 기능

사용자가 다른 언어에서 처럼 단계적인 알고리즘을 적용하는 프로시저를 정의할 수도 있고, 또는 간단히 함수를 정의하여 사용할 수도 있다. 자료의 종류(data type)을 지정해 주지 않아도 좋으며 함수로 정의하는 경우에는 프로그래밍이 다른 언어와 비교할 때 매우 간단해 진다. 예를 들면, 함수 f[n]을 소수를 작은 것부터 n개 나열하는 함수로 정의하고(In[6]), 10개의 소수를 나열하도록 f[10]을 요구하면(In[7]), {2, 3, ... , 29}의 list를 출력한다(Out[7]).

```
In[6]:= f[n_] := Table[Prime[i], {i, n}];
```

In[7]:= f[10]
Out[7]= {2, 3, 5, 7, 11, 13, 17, 19, 23, 29}
일련의 프로그램으로 패키지를 만들 수도 있다.

4. Mathematica와 수학

Mathematica는 Kernel이라 불리우는 기본 프로그램과 영역별 프로그램들이 package의 형태로 구성되어 있다. 현재 IBM을 위한 Window 용 프로그램 (version 2.2)의 package 안에 file형태로 포함되어 있는 영역으로는 대수, 정수론, 그래픽, 통계, 수치해석, 미적분 등이 있다.

Mathematica에는 미적분 관련 함수를 모두 포함하고 있다. 부분미분, 전미분, 부정적분과 정적분 모두를 다룰 수 있으며 Bessel함수나 Gamma 함수 같은 특별한 형태의 함수도 적분할 수 있다.

곡선의 기울기, 다항함수, 유리함수, 삼각함수 등의 그래프, 함수의 전미분, 편미분, 정적분, 부정적분, 이중적분, 중간값정리, 극한, 함수 $z = f(x, y)$의 3차원의 그래프와 등위곡선에 이르기까지 미적분의 전 범위에 걸쳐 Mathematica를 사용할 수 있다. 미분(D[ ]), 부정적분(Integrate[ ]), 또는 2-, 3-차원의 그래프(Plot[ ], Plot3D[ ]) 같이 자주 사용되는 내용은 함수의 형태로 기본 프로그램 안에 내장시켜 놓아 어느 시점에서나 실행이 될 수 있으며, 음함수 미분(Implicit[ ])이나 동화상 기능(Animation[ ]) 같이 상대적으로 가끔 쓰이는 함수들은 미적분, 선형대수, 이산수학, 기하, 미적분 등 영역별로 package 안에 넣어 두고 필요할 때에 실행시키도록 되어 있다.[2)]

2) 모든 기능을 기본으로 해 놓으면 사용에는 편리하지만, 간단한 명령이라도 일단 한 가지 명령만 실행할 때도 기본 package 전체가 kernel로 컴퓨터의 메모리(RAM)에 실리게 되므로 작업용량을 너무 많이 차지하게 된다. 실제로 여러 package에 기타 프로그램을 영역별로 분산시켜 저장하고 있음에도 불구하고 kernel을 초기화 하는데만 1 Mega Byte 이상의 용량을 초과한다.

III. Mathematica를 통한 미적분 학습활동

1. Mathematica의 활용 원칙

Mathematica를 사용하면 II장에서 서술한 바와 같이 이미 만들어진 함수(built-in function)를 이용하여 그래프나, 수 값 또는 수식으로 나타내진 결과들을 원하는 그래프를 (컴퓨터의 memory가 허락하는 한) 얼마든지 신속하게 그릴 수 있고 복잡한 행렬계산이나 미적분 결과를 쉽게 얻을 수 있다. 그러나 이렇게 내장된 함수를 통해 원하는 결과를 손쉽게 구했다고 할 때, 이것이 미적분의 개념이나 원리의 이해, 기술의 학습에 도움이 되겠는가? 만일 공학을 목적으로 답을 구하려고 한다면 이러한 프로그램이 유익하겠지만, 미적분학의 교육이라는 관점에서 본다면 Mathematica의 활용이 어떤 점에서 유익하겠는가? 그러므로 미적분학의 교수·학습에 효과적으로 Mathematica를 활용하려면 이에 관한 기본 원칙을 설정해야 할 것이다.

(1) Mathematica는 수학적 개념의 이해를 위해 사용되어야 한다. 계산기능이 뛰어 나고 표현방식이 다양하기 때문에 지필환경에서는 설명하기 어려운 개념들을 쉽게 설명해 낼 수 있기 때문이다.

(2) 시각화의 장점을 활용하여 문제해결이나 수학적 사고의 확장을 돕는데 Mathematica를 사용할 수 있다. z = f(x, y)와 같이 3차원 공간에서라야 그림이 그려 지는 함수들을 시각화함으로써 평면에서만 그려지던 그래프의 개념을 확장하고 이러한 함수와 관련된 문제해결을 도울 수 있다.

(3) Mathematica의 활용은 학습자로 하여금

미적분의 실생활의 적용가능성에 긍정적인 태도와 인식을 갖도록 할 수 있다. 이는 전통적인 지필환경에서는 다루어 질 수 없었던 수치 등이 관련된 복잡한 상황들의 문제들을 해결할 수 있기 때문이다.

또 이미 만들어진 함수들 이외에도 사용자가 적절히 함수를 정의할 수 있기 때문에 Mathematica의 활용의 지평은 계속 확장될 수 있다.

2. Mathematica를 활용한 미적분활동

(1) 접선의 방정식

점 (-2, 4)에서 $y=x^2$에 그은 접선의 식을, Fermat와 같이, (-2, 4)와 곡선 위의 임의의 점을 연결하는 할선(secant line)의 식을 구하여 보고 그 임의의 점을 (-2, 4)에 무한히 접근시킴으로 구하였다. Mathematica를 활용하여 할선의 극한으로의 접선의 의미를 역동적으로 보여 줄 수 있다.

다음은 그래프의 접선의 기울기를 할선의 극한값으로 구하는 Mathematica의 화면이다[3]. (-2, 4)와 (a, a^2)를 지나는 할선의 기울기를 구하되 a의 값을 -1에서 시작하여 0.1씩 감소시켜 -1.9에 이르기 까지 기울기를 구하여 그 값이 -3부터 -3.9로 변화함을 표로 확인하고 ([Out 1]), a가 -2에 가까워질 때 할선의 기울기의 극한값으로 -4를 구하였다([Out2]). Out[3]은 $x=b$일 때, 즉 $y=x^2$위의 점(b, b^2)에서 곡선에 그은 접선의 기울기이다.

```
In[1]:=
        f[x_]:=x^2;
        Clear[a,m];
        m=(f[a]-f[-2])/(a-(-2));
        TableForm[
            Table[{a,m}, {a, -1.0,-1.9, -0.1}],
            TableHeadings ->
             {None, {"a\n","secant\nslope\n"}}
            ]
Out[1]//TableForm =
        a       secant
                 slope
        -1.     -3.
        -1.1    -3.1
        -1.2    -3.2
        -1.3    -3.3
        -1.4    -3.4
        -1.5    -3.5
        -1.6    -3.6
        -1.7    -3.7
        -1.8    -3.8
        -1.9    -3.9

In[2]:=
        Clear[a];
        Limit[(f[a]-f[-2])/(a-(-2)),a->-2]
Out[2]=
        -4
In[3]:=
        Clear[a];
        Limit[(f[a]-f[-2])/(a-(-2)), a->b]
Out[3]=
        2 b
```

이 할선들을 다음과 같은 명령으로 일련의 그래프로 나타낼 수도 있다.

```
Clear[b]
tanLine[x_,b_]:= 2 b*(x - b) +b^2;
Do[
        Plot[
        {f, tanLine}, {x,-3,3},
        PlotStyle->{{Thickness[0.02],
```

3) In[ ]으로 시작되는 부분은 사용자가 써 넣은 부분이고, Out[ ]으로 시작되는 부분은 컴퓨터가 실행하여 결과로 출력해 낸 부분이다.

```
                GrayLevel[0.5]}},
        PlotRange->{0,9},
        Prolog->{PointSize[.03],
                Point[{b,b^2}]},
        ],
        {b,-2, 2, 0.5}
        ];
```

(2) 정적분

$y=f(x)$ (a≤x≤b)의 정적분은 보통 Riemann 합 ($s_k = \sum_{k=1}^{n} f(c_k)\cdot\triangle x_k$)의 극한으로 구한다. c_k의 값을 구간의 어느 점으로 하는가에 따라 s_k의 값이 달라지며, s_k의 극한 값이 c_k를 잡는 방법에 관계없이 일정한 값에 가까워질 때, 그 극한값이 Riemann적분값이 되는 것이다.

다음은 c_k를 구간의 왼쪽, 오른쪽 경계의 값, 그리고 두 값의 평균값으로 했을 때 각각의 리만합(s_k)을 left-, right-RieamannSum, 그리고 midPointSum이란 이름으로 구하도록 하는 Mathematica 프로그램이다.

먼저 세 함수를 각기 정의하고, f(x)= x^2 으로 했을 때, 1≤x≤2 구간을 5, 8, 500등분($n=5$, 8, 500)한 각각의 경우에 세 가지 리만합을 구하도록 하였다.

```
In[1]:=

        leftRiemannSum[f_,a_,b_,n_]:=
              Block[{h,x,i},
                    h = (b-a)/n;
                    x[i_]:= a + h*i;
                    N[Sum[f[x[i]]*h,{i,0,n-1}]]
                    ];
        rightRiemannSum[f_,a_,b_,n_]:=
              Block[{h,x,i},
                    h = (b-a)/n;
                    x[i_]:= a + h*i;
                    N[Sum[f[x[i]]*h, {i,1,n}]]
                    ];
        midPointSum[f_,a_,b_,n_]:=
                    Block[{h,x,i},
                    h = (b-a)/n;
                    x[i_]:= a + h*i + h/2;
                    N[Sum[f[x[i]]*h,{i,0,n-1}]]
                    ];

        f[x_]:=x^2;
        leftRiemannSum[f,0,2,5]
        rightRiemannSum[f,0,2,5]
        midPointSum[f,0,2,5]
        leftRiemannSum[f,0,2,8]
        rightRiemannSum[f,0,2,8]
        midPointSum[f,0,2,8]
        leftRiemannSum[f,0,2,500]
        rightRiemannSum[f,0,2,500]
        midPointSum[f,0,2,500]

Out[14] = 1.92
Out[15] = 3.52
Out[16] = 2.64
Out[17] = 2.1875
Out[18] = 3.1875
Out[19] = 2.65625
Out[20] = 2.65867
Out[21] = 2.67467
Out[22] = 2.66666
```

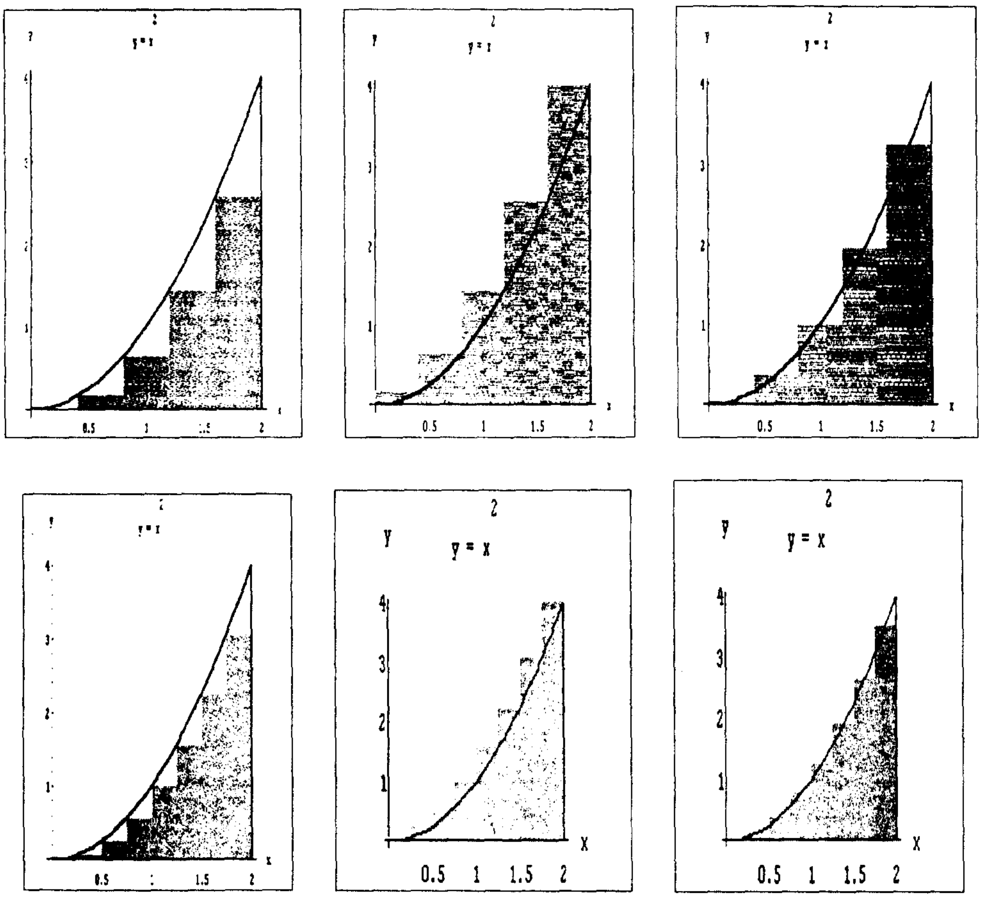
위의 결과에서 구간을 5등분한 경우([Out14-16])와 8등분한 경우([Out17-19])에 세 가지 리만합은 1.9에서 3.5까지, 2.2에서 3.2 까지 차이가 많았으나, 500등분한 경우[Out20-22]에는 2.658에서 2.675사이의 값을 가지므로 차이가 거의 없음을 보여 준다. 물론 이같은 일련의 결과들을 Table[ ]을 이용하여 표로 나타낼 수 있다.

Mathematica는 이 문제 상황을 간단한 프로

그래을 작성함으로써 그래프로 나타내 준다. 다음 프로그램은 구간을 2등분한 경우의 leftRiemannSum의 그래프를 그리는 것이다.

```
In[47]:= region = Plot[
                f[x], {x,0,2},
                Epilog->
                    {Line[{{2,0},{2,4}}]},
                AxesLabel->{" x","y"},
                PlotLabel->"y =" f[x]
        Show[
                region,
                Prolog->{{
                GrayLevel[.5],
                Rectangle[{0,0},{1,f[0]}],
                Rectangle[{1,0},{2,f[1]}]
                }}
                ];
```

다음은 Mathematica로 그린 그래프들로서, $y=x^2(0\leq x\leq 2)$에서 구간을 5, 8 등분했을 때 세가지 Riemann합을 각각 나타낸 것이다.



이 그래프를 통해서 Riemann합의 존재를 직관적으로 알 수 있게 된다.

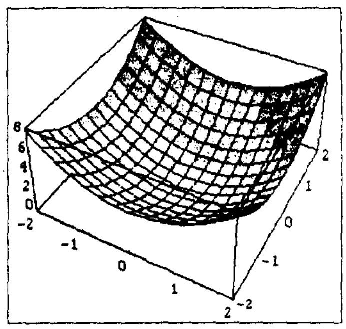
3) 그래프의 활용

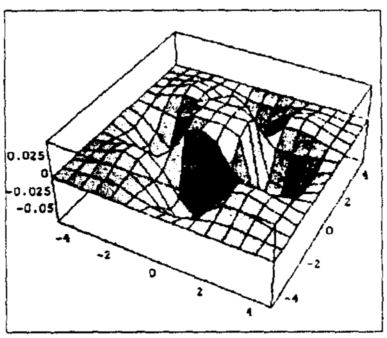
학생들은 주로 평면에 그려진 그래프를 보아왔기 때문에, 독립변수가 두 개인 함수의 그래프에 익숙하지 않다. 3차원에 그려지는 함수의 그래프는 다변수 함수에서 도입되는 편미분 등의 개념을 이해하는 데에도 도움을 줄 수가 있다. 다음은 각각 Mathematica가 그려낸

$$f(x,y) = x^2 + y^2$$

$$f(x,y) = \cos x \cos y \ e^{-\sqrt{x^2+y^2}}$$

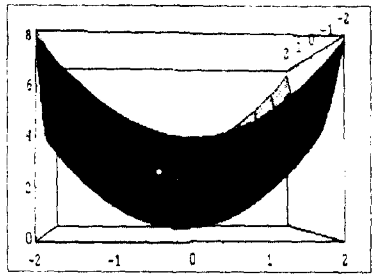
의 그래프이다.





위의 두 그래프를 위해서 Plot3D[x^2 + y^2, {x. -2, 2}, {y, -2, 2}] 와 Plot3D[Cos[x] Cos[y] Exp[- Sqrt[x^2 + y^2]], {x, - 3 Pi/2, 3 Pi/2}, {y, - 3 Pi/2, 3 Pi/2}]를 각각 입력하였다.

여기에 옵션으로 시점을 변화시키면 그래프 모양을 바꿀 수가 있다. 예를 들면 그래프 $f(x,y) = x^2 + y^2$의 경우에 ViewPoint라는 옵션을 주면 다음과 같은 그래프를 얻을 수 있다.



Plot3D[x^2 + y^2, {x. -2, 2}, {y, -2, 2}, ViewPoint -> {-0.032, -3.383, -0.078}]

이를 통하여 y값을 고정시킬 때, x의 변화에 따른 f(x,y)의 변화율을 직관적으로도 알 수 있어서, 편미분 $\frac{\partial f}{\partial x}$의 의미를 확실히 이해하고 이를 사용할 수 있게 된다.

IV. 요약 및 결론

컴퓨터는 수치해석과 같이 계산이 크게 복잡하여 과거에는 수학의 영역에서 다루기 힘든 내용들을 수학의 영역에서 다룰 수 있도록 하였다. 컴퓨터의 등장으로 수학의 새로운 영역들이 강조되고 생겨나서 수학적 활동을 바꾼다.

Mathematica는 대학의 미적분학의 강의에 적합하게 개발된 프로그램으로 중고등학교에서도 사용가능하다. Mathematica는 개념이나 원리의 실연에 이용될 수 있으며 계산기능 외에 시각적 기능이 뛰어나므로 연역적으로 규칙에 의존한 것 같은 수학에서 의미있는 수학으로

강의를 변화시킬 수 있다. 또 간단한 프로그램을 통해 알고리즘의 습득도 가능해 지며 문제해결의 상황을 실생활에 가깝게 함으로써 수학의 활용에도 기여 할 수 있게 된다.

Mathematica를 강의에 적극적으로 활용하기 위해서는 한 강의의 수강자 모두가 어려움없이 컴퓨터를 사용할 수 있는 물리적 환경이 먼저 만들어져야 한다. PC를 이용한다고 하면 Mathematica를 사용하는데 무리가 없도록 적어도 16 MegaByte의 RAM과 수학 coprocessor를 반드시 내장하고 있어야 한다. 그렇지 않은 경우에는 프로그램이 memory부족으로 실행되지 않거나, 실행되는 도중에 컴퓨터가 down되기 때문이다. 또한 여러 영역에서 학습에 도움이 되도록 프로그램을 작성하고 자료는 모으는 일들이 중요하다.

참 고 문 헌

황일 외(역) (1995). 미분적분학과 해석기하학. 서울: 대영사. (Thomas, G. & Finney, R. (1992). Calculus and analytic geometry. 8th ed. Reading. MA: Addison-Wesley Pub. Co.)

Bullough, R. & Beatty, L. (1987). Classroom applications of microcomputers. Chapter. 4. Columbus. OH: Merrill Publishing Co.

Cornu, B. (1991). The computer: Some changes in mathematics teaching and learning. In D. Ferguson. (ed.), Advanced educational technologies for mathematics and science. 687-708. New York: Springer - Verlag.

Finch, J. & Lehmann, M. (1992). Exploring calculus with Mathematica®. Reading. MA: Addison-Wesley Publishing Co.

Wolfram, S. (1991). Mathematica: A system for doing mathematics by cmputer. 2nd ed. Reading, MA: Addison-Wesley Publishing