

자유 주제 탐구 활동

- 1. 활동 목적: 학업 역량과 전공(계열)적합성 강화를 통해 나의 대입을 성공으로 이끌기!
- 2. 활동 방법: 1학년 시기부터 3학년 1학기동안 학습하거나 탐구했던 내용에서 영감을 얻어 해당 주제에 대해 스스로 심화 탐구를 한 후 그 결과를 정리하여 산출물을 제작하고, 발표하는 활동
- 3. 활동 일정

회차	날짜	일정	회차	날짜	일정
1	4월 21일	활동 안내 및 주제 선정	2	4월 28일	주제 선정 및 개별 피드백
3	5월 12일	탐구 1차시(활동지 작성)	4	5월 19일	탐구 2차시(활동지 작성)
5	5월 19일	탐구 3차시(결과 정리), 발표순서 결정	6	6월 2일	탐구 결과 발표
7	6월 9일	탐구 결과 발표	8	6월 16일	탐구 결과 발표
9	6월 23일	활동 최종 정리 및 소감문 작성			

4. 예시 주제

- 진로가 경제학 연구원이라 2학년 때 한국지리 시간에 ‘젠트리피케이션’에 대해서 탐구한 적이 있었음. 젠트리피케이션의 해결방안으로 ‘임대차 보호법’을 제시했었는데 신문 기사에서 ‘임대차 보호법의 허와 실’을 접한 뒤 임대차 보호법의 한계와 개선 방안을 탐구하여 발표함. → **기준에 탐구한 주제 및 해결방안을 보다 비판적으로 생각해 더 좋은 방향을 탐구한 사례**
- 진로가 물리학 관련 분야였는데 최근 ‘전자물리학’ 분야에 관심이 생겨 다이오드에 관련된 내용을 2학년 물리 시간에 학습하고 탐구한 적이 있음. 이를 바탕으로 다이오드를 활용한 정류 회로, 역방향 바이어스에서 항복 전압에 의한 다이오드의 파괴 과정에 대해 궁금증이 생겨 해당 내용을 탐구하여 발표함. → **희망 전공 분야의 세부 주제를 선정한 후 그 세부 주제에 담긴 원리를 깊이있게 탐구한 사례**
- 진로가 미디어 분야라서 2학년 정치와 법 교과 시간에 같은 내용의 사실을 서로 다른 시각으로 보도하는 기사를 찾아 읽던 중 미디어 리터러시의 필요성을 알게됨. 이후 이에 대해 알아보고자 미디어 리터러시 관련 설문 및 문헌 연구를 진행했고, 더 확장시켜 현재 우리나라에서 미디어 리터러시 교육이 얼마나 이루어지고 있는지에 알아보고자 ‘우리나라 미디어 리터러시 교육의 현황과 앞으로의 발전 방향’이라는 주제로 발표함. → **기준에 탐구한 주제를 확장시켜 주변 사람들이 해당 주제에 대해 어떻게 인식하고 있는지를 자료 조사하는 등 현장 연구에 적용한 사례**
- 진로가 화장품 연구원이라 2학년 화학 시간에 ‘산호초의 백화현상과 같은 화장품의 자연 환경 파괴 사례’에 대해서 관심이 생겨 탐구한 적이 있음. 이를 더욱 확장하고자 화장품이 환경 뿐만 아니라 인체에 미치는 영향을 분석하고, 대표적인 화장품 속 유해성분을 탐구하고, 관련 정책으로 ‘전성분 표시제’를 연구해 발표함. → **기준에 탐구한 주제를 확장시켜 이와 비슷한 수준의 또다른 주제를 탐구한 사례**
- 자신이 가장 좋아하는 물리, 수학 분야와 관련된 지식을 탐구하고자 함. 특히 물리적인 현상을 수학적으로 해석하는 역량을 기르고자 ‘벡터 미적분학’ 관련 내용을 탐구해 친구들에게 알기 쉬운 수준으로 해석하여 발표함. → **전공과 관련된 기초 학문적 지식(수학, 물리 등)을 탐구해 자신만의 방법으로 이해한 후 발표한 사례**

★ 기존의 탐구 내용에 대해 반대되는 의견을 탐구해 해당 주제에 대한 폭넓은 인식을 보여주는 발표도 좋음.

★ 주제 선정에 있어서는 선생님께 적극적으로 피드백 받을 것.

★ 이과 계열이라면 수학, 물리 등 기초 교과 관련 주제 탐구도 좋으며 탐구 과정에서 의미있는 책을 활용해보는 것도 좋음!

- 5. 내가 탐구할 주제 정하기 → 생활기록부에서 자주 등장하는 개념을 나열해보고, 그 개념을 중심으로 발전시켜나가 보세요!

2025학년도 1학기 3학년 반 <자유 주제 탐구 활동> 활동지							
희망 진로	인공지능	발표 (제출) 날짜	7.6	학번	3112	이름	송정아
선정 주제	파이썬과 행렬 연산을 활용한 벡터의 시각화						
해당 주제를 선정한 이유	작년 기하 교과 독서 탐구 활동에서 인공지능 수학 관련 도서를 읽었는데, 그중 행렬 연산의 코딩에 대해 다룬 내용이 인상 깊었다. 이를 계기로 파이썬을 통해 행렬 연산을 구현하고, 그 연산이 벡터에 어떤 영향을 주는지 시각적으로 확인해 보고자 이 주제를 선정하였다.						
세부 탐구 내용	※ 탐구 내용의 서론-본론-결론을 대략적으로 갖춘 후 개조식으로 작성할 것						
	<div><서론></div> <div>1. 연구 목적</div> <p>파이썬을 이용해 행렬 연산을 벡터에 적용한 결과를 시각화하고, 행렬이 벡터의 방향과 위치에 어떤 변화를 주는지 확인한다.</p> <div>2. 이론적 배경</div> <p>행렬이란 수나 식을 직사각형 모양으로 배열한 것으로, 회전 행렬은 유클리드 공간에서 원점을 중심으로 회전변환을 하게 하는 행렬을 말한다. 회전 행렬은 특별 직교 행렬에 속해 벡터의 크기를 보존한다는 성질을 가지고 있다.</p> <p>2차원 유클리드 공간인 경우에 원점을 중심으로 시계 반대 방향으로 시계 반대 방향으로 θ만큼 회전한 회전 행렬은 다음과 같다.</p> <div>$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$</div> <p>이 행렬을 벡터에 곱하면 벡터는 원점을 중심으로 θ만큼 회전하게 된다. 또한 회전 행렬은 직교성에 의해 벡터의 크기를 보존한다는 성질을 가지고 있다.</p> <div><본론></div> <div>1. 연구 문제</div> <p>회전 행렬을 파이썬 코드로 구현했을 때, 벡터의 방향이나 위치가 어떻게 달라지는가?</p> <div>2. 연구 방법</div> <p>2-1. Python의 numpy 라이브러리를 사용해 벡터와 회전 행렬을 정의</p> <p>2-2. matplotlib을 이용해 회전 전후의 벡터를 시각화</p> <p>2-3. 회전 각도 45도로 설정</p> <p>2-4. 단위 벡터 (1,0), (0,1)를 사용하여 효과 명확화</p> <div>3. 연구 결과</div> <p>아래 사진과 같이 회전 행렬을 파이썬 코드로 구현하고 시각화한 결과, 그래프상에서 빨간색과 초록색 벡터는 각각 x축과 y축을 나타냈고, 파란색과 보라색 벡터는 45도 회전된 방향을 나타냈다. 이를 통해 벡터의 길이는 유지되면서 방향만 바뀌었음을 확인할 수 있었으며, 이는 회전 행렬이 벡터의 크기는 보존하면서 방향만 변화시킨다는 이론과 일치한다.</p>						

<div data-bbox="114 887 220 954" data-label="Text"><p>세부 탐구 내용</p></div>	<div data-bbox="264 161 1011 512" data-label="Code-Block"><pre>Windows PowerShell PS C:\Users\W66성아\Desktop> python Python 3.13.3 (tags/v9.13.3:8280bb5, Apr 8 2025, 14:47:33) [AMD64] on win32 Type "help", "copyright", "credits" or "license()" for more information. >>> import numpy as np >>> import matplotlib.pyplot as plt >>> vectors = np.array([[1, 0], [0, 1]]) >>> theta = np.radians(45) >>> rotation_matrix = np.array([[np.cos(theta), -np.sin(theta)], [np.sin(theta), np.cos(theta)]]) >>> rotated_vectors = vectors @ rotation_matrix.T >>> fig, ax = plt.subplots() >>> ax.quiver(0, 0, vectors[0, 0], vectors[0, 1], color='r', angles='xy', scale_units='xy', scale=1, label='기존 x축') >>> ax.quiver(0, 0, vectors[1, 0], vectors[1, 1], color='g', angles='xy', scale_units='xy', scale=1, label='기존 y축') >>> ax.quiver(0, 0, rotated_vectors[0, 0], rotated_vectors[0, 1], color='b', angles='xy', scale_units='xy', scale=1, label='회전된 x축') >>> ax.quiver(0, 0, rotated_vectors[1, 0], rotated_vectors[1, 1], color='purple', angles='xy', scale_units='xy', scale=1, label='회전된 y축') >>> ax.set_xlim(-1.5, 1.5) >>> ax.set_ylim(-1.5, 1.5) >>> ax.set_aspect('equal') >>> plt.grid() >>> plt.legend() >>> plt.title("회전 행렬 적용 전후의 벡터") >>> text(0.5, 1.0, "회전 행렬 적용 전후의 벡터") >>> plt.show()</pre></div> <div data-bbox="264 551 411 582" data-label="Section-Header"><p>파이썬 코드</p></div> <div data-bbox="284 624 873 1124" data-label="Figure"></div> <div data-bbox="264 1162 547 1193" data-label="Section-Header"><p>코드 적용 결과 그래프</p></div> <div data-bbox="264 1249 373 1283" data-label="Section-Header"><p><결론></p></div> <div data-bbox="300 1299 651 1335" data-label="Section-Header"><p>연구 결과 요약 및 논의</p></div> <div data-bbox="264 1350 1509 1675" data-label="Text"><p>이번 탐구에서는 파이썬으로 회전 행렬을 구현하고, 이를 단위 벡터인 (1, 0)과 (0, 1)에 적용하여 벡터의 회전 결과를 시각화하였다. 벡터의 길이(크기)는 유지되고 방향만 변화한 것을 확인할 수 있었으며, 이는 회전 행렬의 성질과 일치하였다.</p><p>또한, 이 과정을 시각화함으로써 행렬의 곱이 단순 계산을 넘어, 공간 변화를 어떻게 나타내는지 직관적으로 이해할 수 있었다. 이러한 행렬의 기하학적 성질은 데이터를 벡터로 변환하거나, 특징 공간에서의 방향성 변환 등 인공지능 분야에서 매우 유용하게 활용될 수 있다.</p></div> <tr><td data-bbox="82 1675 252 1874"><div data-bbox="114 1731 220 1821" data-label="Text"><p>근거자료 (책, 신문기사 등)</p></div></td><td data-bbox="252 1675 1522 1874"><div data-bbox="379 1753 1398 1798" data-label="Text"><p>모두의 인공지능 기초 수학(서지영), 네이버 지식백과, 챗GPT, Claude</p></div></td></tr>	<div data-bbox="114 1731 220 1821" data-label="Text"><p>근거자료 (책, 신문기사 등)</p></div>	<div data-bbox="379 1753 1398 1798" data-label="Text"><p>모두의 인공지능 기초 수학(서지영), 네이버 지식백과, 챗GPT, Claude</p></div>
<div data-bbox="114 1731 220 1821" data-label="Text"><p>근거자료 (책, 신문기사 등)</p></div>	<div data-bbox="379 1753 1398 1798" data-label="Text"><p>모두의 인공지능 기초 수학(서지영), 네이버 지식백과, 챗GPT, Claude</p></div>		