

대분류/08
문화·예술·디자인·방송

중분류/03
문화콘텐츠

소분류/02
문화콘텐츠제작

세분류/05
게임콘텐츠제작

능력단위/34, 42

NCS학습모듈

게임 네트워크와 클라이언트 프로그래밍

LM0803020534_18v4
LM0803020542_18v1



교육부

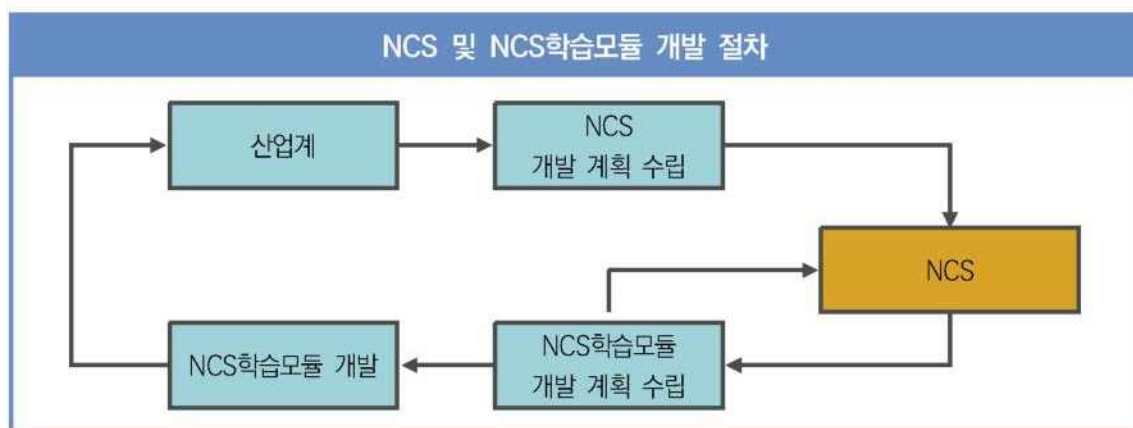
NCS 학습모듈은 교육훈련기관에서 출처를 명시하고 교육적 목적으로 활용할 수 있습니다. 다만 NCS 학습모듈에는 국가(교육부)가 저작권 일체를 보유하지 않은 저작물들(출처가 표기되어 있는 도표, 사진, 삽화, 도면 등)이 포함되어 있으므로 이러한 저작물들의 변형, 복제, 공연, 배포, 공중 송신 등과 이러한 저작물들을 활용한 2차 저작물의 생성을 위해서는 반드시 원작자의 동의를 받아야 합니다.

NCS학습모듈의 이해

※ 본 NCS학습모듈은 「NCS 국가직무능력표준」사이트(<http://www.ncs.go.kr>) 에서 확인 및 다운로드할 수 있습니다.

I NCS학습모듈이란?

- 국가직무능력표준(NCS: National Competency Standards)이란 산업현장에서 직무를 수행하기 위해 요구되는 지식·기술·소양 등의 내용을 국가가 산업부문별·수준별로 체계화한 것으로 산업현장의 직무를 성공적으로 수행하기 위해 필요한 능력(지식, 기술, 태도)을 국가적 차원에서 표준화한 것을 의미합니다.
- 국가직무능력표준(이하 NCS)이 현장의 ‘직무 요구서’라고 한다면, **NCS학습모듈은 NCS의 능력단위를 교육훈련에서 학습할 수 있도록 구성한 ‘교수·학습 자료’입니다.** NCS학습모듈은 구체적 직무를 학습할 수 있도록 이론 및 실습과 관련된 내용을 상세하게 제시하고 있습니다.

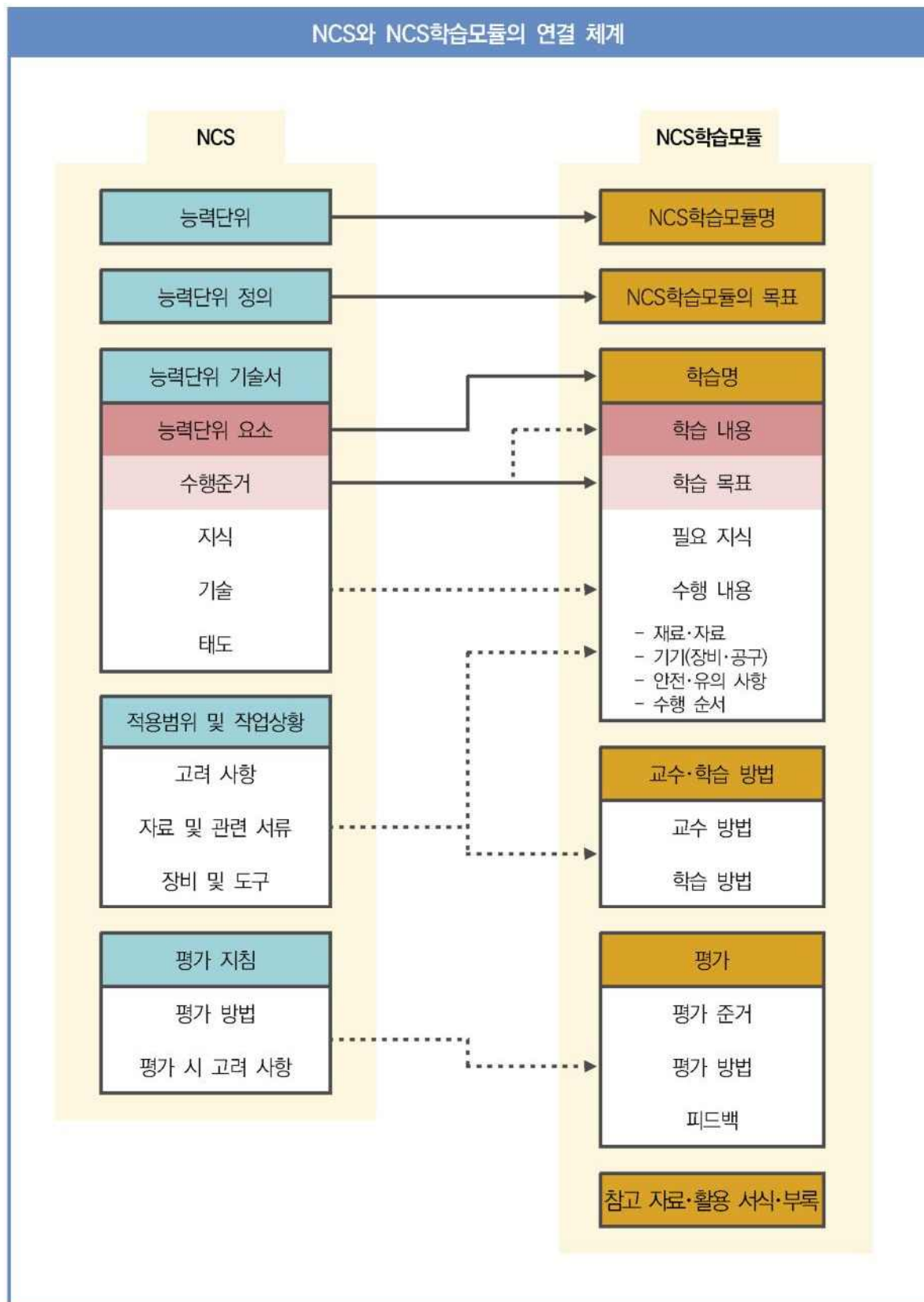


- **NCS학습모듈은 다음과 같은 특징을 가지고 있습니다.**

첫째, NCS학습모듈은 산업계에서 요구하는 직무능력을 교육훈련 현장에 활용할 수 있도록 성취목표와 학습의 방향을 명확히 제시하는 가이드라인의 역할을 합니다.

둘째, NCS학습모듈은 특성하고, 마이스터고, 전문대학, 4년제 대학교의 교육기관 및 훈련기관, 직장교육기관 등에서 표준교재로 활용할 수 있으며 교육과정 개편 시에도 유용하게 참고할 수 있습니다.

○ NCS와 NCS학습모듈 간의 연결 체계를 살펴보면 아래 그림과 같습니다.



II NCS학습모듈의 체계

○ NCS학습모듈은 1. NCS학습모듈의 위치, 2. NCS학습모듈의 개요, 3. NCS학습모듈의 내용 체계, 4. 참고 자료, 5. 활용서식/부록으로 구성되어 있습니다.

1. NCS학습모듈의 위치

○ NCS학습모듈의 위치는 NCS 분류 체계에서 해당 학습모듈이 어디에 위치하는지를 한 눈에 볼 수 있도록 그림으로 제시한 것입니다.

[NCS-학습모듈의 위치]		
대분류	문화·예술·디자인·방송	
중분류	문화콘텐츠	
소분류	문화콘텐츠제작	
세분류		
방송콘텐츠제작	능력단위	학습모듈명
영화콘텐츠제작	프로그램 기획	프로그램 기획
음악콘텐츠제작	아이템 선정	아이템 선정
광고콘텐츠제작	자료 조사	자료 조사
게임콘텐츠제작	프로그램 구성	프로그램 구성
애니메이션 콘텐츠제작	캐스팅	캐스팅
만화콘텐츠제작	제작계획	제작계획
캐릭터제작	방송 미술 준비	방송 미술 준비
스마트문화앱 콘텐츠제작	방송 리허설	방송 리허설
영사	야외촬영	야외촬영
	스튜디오 제작	스튜디오 제작

학습모듈은

NCS 능력단위 1개당 1개의 학습모듈 개발을 원칙으로 합니다. 그러나 필요에 따라 고용단위 및 교과단위를 고려하여 능력단위 몇 개를 묶어 1개 학습모듈로 개발할 수 있으며, NCS 능력단위 1개를 여러 개의 학습모듈로 나누어 개발할 수도 있습니다.

2. NCS학습모듈의 개요

○ NCS학습모듈의 개요는 학습모듈이 포함하고 있는 내용을 개략적으로 설명한 것으로

학습모듈의 목표, 선수학습, 학습모듈의 내용 체계, 핵심 용어 로 구성되어 있습니다.

학습모듈의 목표	해당 NCS 능력단위의 정의를 토대로 학습 목표를 작성한 것입니다.
선수학습	해당 학습모듈에 대한 효과적인 교수·학습을 위하여 사전에 이수해야 하는 학습모듈, 학습 내용, 관련 교과목 등을 기술한 것입니다.
학습모듈의 내용 체계	해당 NCS 능력단위요소가 학습모듈에서 구조화된 체계를 제시한 것입니다.
핵심 용어	해당 학습모듈의 학습 내용, 수행 내용, 설비·기자재 등 가운데 핵심적인 용어를 제시한 것입니다.

제작계획 학습모듈의 개요

학습모듈의 목표

본격적인 촬영을 준비하는 단계로서, 촬영 대본을 확정하고 제작 스태프를 조직하며 촬영 장비와 촬영 소품을 준비할 수 있다.

선수학습

제작 준비(LM0803020105_13v1), 섭외 및 제작스태프 구성(LM0803020104_13v1), 촬영 제작(LM0803020106_13v1), 촬영 장비 준비(LM0803040204_13v1.4), 미술 디자인 협의하기(LM0803040203_13v1.4)

학습모듈의 내용체계

학습	학습 내용	NCS 능력단위 요소	
		코드번호	요소 명칭
1. 촬영 대본 확정하기	1-1. 촬영 구성안 검토와 수정	0803020114_16.3.1	촬영 대본 확정하기
2. 제작 스태프 조직하기	2-1. 기술 스태프 조직	0803020114_16.3.2	제작 스태프 조직하기
	2-2. 미술 스태프 조직		
	2-3. 전문 스태프 조직		
3. 촬영 장비 계획하기	3-1. 촬영 장비 점검과 준비	0803020114_16.3.3	촬영 장비 계획하기
4. 촬영 소품 계획하기	4-1. 촬영 소품 목록 작성	0803020114_16.3.4	촬영 소품 계획하기
	4-2. 촬영 소품 제작 의뢰		

핵심 용어

촬영 구성안, 제작 스태프, 촬영 장비, 촬영 소품

학습모듈의 목표는

학습자가 해당 학습모듈을 통해 성취해야 할 목표를 제시한 것으로, 교수자는 학습자가 학습모듈의 전체적인 내용흐름을 파악하도록 지도할 수 있습니다.

선수학습은

교수자 또는 학습자가 해당 학습모듈을 교수·학습하기 이전에 이수해야 하는 교과목 또는 학습모듈(NCS 능력단위) 등을 표기한 것입니다. 따라서 교수자는 학습자가 개별 학습, 자기 주도 학습, 방과 후 활동 등 다양한 방법을 통해 이수할 수 있도록 지도하는 것을 권장합니다.

핵심 용어는

해당 학습모듈을 대표하는 주요 용어입니다. 학습자가 해당 학습모듈을 통해 학습하고 평가받게 될 주요 내용을 알 수 있습니다. 「NCS 국가직무능력표준」 사이트(www.ncs.go.kr)의 색인(찾아보기) 중 하나로 이용할 수 있습니다.

3. NCS학습모듈의 내용 체계

○ NCS학습모듈의 내용은 크게 **학습**, **학습 내용**, **교수·학습 방법**, **평가**로 구성되어 있습니다.

학습	해당 NCS 능력단위요소 명칭을 사용하여 제시한 것입니다. 학습은 크게 학습 내용, 교수·학습 방법, 평가로 구성되며 해당 NCS 능력단위의 능력단위 요소별 지식, 기술, 태도 등을 토대로 내용을 제시한 것입니다.
학습 내용	학습 내용은 학습 목표, 필요 지식, 수행 내용으로 구성되며, 수행 내용은 재료·자료, 기기(장비·공구), 안전·유의 사항, 수행 순서, 수행 tip으로 구성한 것입니다. 학습모듈의 학습 내용은 실제 산업현장에서 이루어지는 업무활동을 표준화된 프로세스에 기반하여 다양한 방식으로 반영한 것입니다.
교수·학습 방법	학습 목표를 성취하기 위한 교수자와 학습자 간, 학습자와 학습자 간 상호 작용이 활발하게 일어날 수 있도록 교수자의 활동 및 교수 전략, 학습자의 활동을 제시한 것입니다.
평가	평가는 해당 학습모듈의 학습 정도를 확인할 수 있는 평가 준거 및 평가 방법, 평가 결과의 피드백 방법을 제시한 것입니다.

학습 1	촬영 대본 확정하기	<p>학습은</p> <p>해당 NCS 능력단위요소 명칭을 사용하여 제시하였습니다. 하나의 학습은 일반교과의 '대단원'에 해당되며, 학습모듈을 구성하는 가장 큰 단위가 됩니다. 또한 하나의 직무를 수행하기 위한 가장 기본적인 단위로 사용할 수 있습니다.</p> <p>학습 내용은</p> <p>NCS 능력단위요소별 수행준거를 기준으로 제시하였습니다. 일반교과의 '중단원'에 해당합니다.</p> <p>학습 목표는</p> <p>학습 내용을 이수할 때 학습자가 갖춰야 할 행동 수준을 의미합니다. 따라서 수업시간의 과목 목표로 활용할 수 있습니다.</p> <p>필요 지식은</p> <p>해당 NCS의 지식을 토대로 학습에 대한 이해와 성과를 제고하기 위해 반드시 알아야 할 주요 지식을 제시하였습니다. 필요 지식은 수행에 꼭 필요한 핵심 내용을 위주로 제시하여 교수자의 역할이 매우 중요하며, 이후 수행 순서와 연계하여 교수·학습으로 진행할 수 있습니다.</p>
학습 2	제작 스태프 조직하기	
학습 3	촬영 장비 계획하기	
학습 4	촬영 소품 계획하기	

2-1. 기술 스태프 조직

학습 목표 • 프로그램 제작에 적합한 기술 스태프를 조직할 수 있다.

필요 지식 /

① 기술 스태프의 구성

프로그램의 장르에 따라 구성하는 기술 스태프는 많은 차이가 있다. 같은 장르의 프로그램이라도 그 형식이나 내용, 규모에 따라서 구성되는 기술 스태프의 종류와 인원 수는 천차만별이다.

1. 스튜디오 프로그램

토크쇼, 종합 구성, 예능과 같은 스튜디오 프로그램은 부조정실과 스튜디오를 사용하여 제작하기 때문에 많은 기술 스태프가 필요하다.

수행 내용 / 기술 스태프 구성표 작성하기

재료·자료

- 방송프로그램 제작 기획서 및 방송 대본, 콘티(continuity), 제작 일정, 운용표
- 장비 및 시설, 제작 시설 배정 의뢰서 및 배정표, 방송 기술 스태프 데이터베이스(DB) 자료

기기(장비·공구)

- 컴퓨터 등

안전·유의 사항

- 프로그램의 내용과 제작 방법을 분석하고, 각 스태프들의 역할을 신중하게 검토한다.

수행 순서

- 1 방송 대본이나 콘티(continuity), 큐 시트를 분석하고, 프로그램의 내용적 특성, 제작 과정에 대한 자료를 수집한다.
- 2 프로그램 제작 방법을 결정한다.
 1. 스튜디오 녹화를 할 것인가, 야외 촬영을 할 것인가 검토한다.

수행 tip

- 스태프의 결정은 스태프 간의 호흡을 중 요시하여 선정해야 프로그램의 질을 향상시킬 수 있다.

수행 내용은

해당 학습모듈에서 제시한 내용 중 기술(skill)을 습득하기 위한 실습과제로 활용할 수 있습니다.

재료·자료는

수행 내용을 수행하는데 필요한 재료 및 준비물로 실습 시 활용할 수 있습니다.

기기(장비·공구)는

수행 내용에 필요한 기본적인 장비 및 도구를 제시하였습니다. 제시된 기기 외에도 수행에 필요한 다양한 도구나 장비를 활용할 수 있습니다.

안전·유의사항은

수행 내용을 수행하는 데 있어 안전상 주의해야 할 점 및 유의사항을 제시하였습니다. 실습 시 유념해야 하며, NCS의 고려 사항도 추가적으로 활용할 수 있습니다.

수행 순서는

실습 과제의 진행 순서로 활용할 수 있습니다.

수행 tip은

수행 내용에서 실습을 용이하게 할 수 있는 아이디어를 제시하였습니다. 수행 tip은 지도상의 안전 및 유의사항 외에 전반적으로 적용되는 주안점 및 수행 과제 목적에 대한 보충설명, 추가사항 등으로 활용할 수 있습니다.

학습2 교수·학습 방법

교수 방법

- 방송 프로그램의 기술적 요소, 미술 구성 요소, 특수 촬영에 대해 설명한다.
- 방송 프로그램 제작에서 각 기술 스태프의 역할에 대해 설명한다.
- 방송 프로그램을 분석하고 필요한 기술 스태프를 구성할 수 있도록 지시한다.

학습 방법

- 방송 프로그램의 기술적 요소, 미술 구성 요소, 특수 촬영에 대해서 알아본다.
- 프로그램 제작에 필요한 기술 스태프의 역할을 이해하고, 기술 스태프 구성표를 작성한다.

교수·학습 방법은

학습 목표를 성취하는 데 필요한 교수 방법과 학습 방법을 제시하였습니다.

교수 방법은

해당 학습 활동에 필요한 학습 내용, 학습 내용과 관련된 자료명, 자료 형태, 수행 내용의 진행 방식 등에 대하여 제시하였습니다. 또한 학습자의 수업참여도 제고 방법 및 수업 진행상 유의사항 등도 제시하였습니다. 선수학습이 필요한 학습을 학습자가 숙지하였는지 교수자가 확인하는 과정으로 활용할 수도 있습니다.

학습 방법은

해당 학습 활동에 필요한 학습자의 자기 주도 학습 방법을 제시하였습니다. 또한 학습자가 숙달해야 할 실기 능력과 학습 과정에서 주의해야 할 사항 등도 제시하였습니다. 학습자가 학습을 이수하기 전 반드시 숙지해야 할 기본 지식을 학습하였는지 스스로 확인하는 과정에 활용할 수 있습니다.

학습2

평 가

평가 준거

- 평가자는 학습자가 학습 목표를 성공적으로 달성하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가해야 한다.

학습 내용	학습 목표	성취수준	상	중	하
기술 스태프 조직	- 프로그램 제작에 적합한 기술 스태프를 조직할 수 있다.				
미술 스태프 조직	- 프로그램 제작에 적합한 미술 스태프를 조직할 수 있다.				
전문 스태프 조직	- 프로그램 특수 촬영을 위한 전문 스태프를 조직할 수 있다.				

평가 방법

- 사례 연구

학습 내용	평가 항목	성취수준	상	중	하
기술 스태프 조직	- 프로그램에서 기술적 요소의 파악 여부 - 기술 스태프의 역할 파악 여부 - 프로그램에 필요한 기술 스태프 구성표 작성 능력				

피드백

1. 사례 연구
 - 프로그램을 선택하여 기술 스태프, 미술 스태프, 전문 스태프 구성표를 제시와 같이 작성하였는지 개인별 능력을 평가한 후, 그 결과를 모든 학습자에게 공유하도록 한다.

평가는

NCS 능력단위의 평가 방법과 평가 시 고려사항을 준용하여 작성합니다. 교수자와 학습자가 평가 항목별 성취수준 확인 시 활용할 수 있습니다.

평가 준거는

학습자가 학습을 어느 정도 성취하였는지 평가하기 위한 기준을 제시하고 있습니다. 학습 목표와 연계하여 단위수업 시간에 평가 항목 별 성취수준을 평가하는 데 활용할 수 있습니다.

평가 방법은

NCS 능력단위의 평가 방법을 참고하였으며, 평가 준거에 따른 평가 방법을 2개 이상 제시합니다. 평가 방법의 종류는 포트폴리오, 문제해결 시나리오, 서술형 시험, 논술형 시험, 사례 연구, 평가자 체크리스트, 작업장 평가 등이 있으며, NCS 능력단위 요소 별 수행 수준을 평가하는 데 가장 적절한 방법을 선정하여 활용할 수 있습니다.

피드백은

평가 후에 학습자들에게 평가 결과를 피드백하여 학습 목표를 달성하는 데 활용할 수 있습니다.

4. 참고 자료

참 고 자 료

- 교육부(2013). 섭외 및 제작스태프 구성(LM0803020104_13v1). 한국직업능력개발원.

참고 자료는

해당 학습모듈에 제시된 인용 자료의 출처를 제시하였습니다. 교수·학습의 과정에서 참고로 활용할 수 있습니다.

5. 활용 서식/부록

활 용 서 식

스튜디오 기술 스태프 구성표

직종	이름	연락처	소속	특이사항	비고
기술감독					
조명감독					

활용 서식은

평가 서식, 실습 시트 등 교수·학습 시 활용할 수 있는 다양한 서식들로 구성하였습니다. 수행에서 평가에 이르기까지 필요한 서식을 해당 모듈의 특성에 맞춰 개발하거나 기존의 양식을 활용하여 제시하였습니다.

부 록

[디지털 텔레비전 방송프로그램 음량 등에 관한 기준]

제정 2014. 11. 29. 마태장초과학부 고시 제2014-87호

제1항 총칙

제1조(목적) 이 고시는 방송법 제70조의2제1항에 따라 방송사업자가 디지털 텔레비전 방송프로그램 및 방송광고의 음량을 일정하게 유지하기 위해 필요한 사항을 규정함을 목적으로 한다.

부록은

활용 서식 이외에 교수·학습 과정에서 참고할 수 있는 자료가 있는 경우 제시하였습니다.

[NCS-학습모듈의 위치]

대분류	문화·예술·디자인·방송
중분류	문화콘텐츠
소분류	문화콘텐츠제작

세분류	능력단위	학습모듈명
방송콘텐츠제작	게임 개발관리	게임 개발관리
영화콘텐츠제작	게임 서비스 운영	게임 서비스 운영
음악콘텐츠제작	게임 사업 기획	게임 사업과 서비스 기획
광고콘텐츠제작	게임 서비스 기획	
게임콘텐츠제작	게임 사전기획	게임 사전기획과 콘셉트 기획
애니메이션콘텐츠 제작	게임 콘셉트 기획	
만화콘텐츠제작	게임 시스템 기획	게임 시스템 기획
캐릭터콘텐츠제작	게임 레벨 기획	게임 레벨과 UI/UX 기획
스마트문화앱콘텐츠 제작	게임 UI/UX 기획	
영사	게임 3D 캐릭터 제작	게임 3D 캐릭터와 배경 제작
완구콘텐츠제작	게임 3D 배경 제작	
드론콘텐츠제작	게임 UI 그래픽디자인	게임 UI와 이펙트 그래픽디자인
	게임 이펙트 디자인	
	게임 배경음악	게임 배경음악과 효과음

게임 효과음	
게임 알고리즘	게임 알고리즘과 게임엔진 프로그래밍
게임엔진 응용 프로그래밍	
게임플랫폼 응용 프로그래밍	게임플랫폼과 UI/UX 프로그래밍
게임 UI/UX 프로그래밍	
게임 인공지능 프로그래밍	게임 인공지능과 그래픽 프로그래밍
게임 그래픽 프로그래밍	
게임 데이터베이스 프로그래밍	게임 데이터베이스와 웹 프로그래밍
게임 웹 프로그래밍	
게임 네트워크 프로그래밍	게임 네트워크와 클라이언트 프로그래밍
게임 클라이언트 프로그래밍	
게임 캐릭터 원화 제작	게임 캐릭터 및 배경 원화 제작
게임 배경 원화 제작	
게임 2D 캐릭터 제작	게임 2D 캐릭터와 배경 제작
게임 2D 배경 제작	
게임 2D 애니메이션 제작	게임 애니메이션 제작
게임 3D 애니메이션 제작	

차 례

학습모듈의 개요	1
학습 1. 게임 서버 시스템 설계하기	
1-1. 게임 서버 시스템 파악	3
1-2. 게임 서버 시스템 설계	6
• 교수 · 학습 방법	9
• 평가	10
학습 2. 서버 네트워크 프로그래밍하기	
2-1. 네트워크 프로그래밍 제작	12
• 교수 · 학습 방법	15
• 평가	16
학습 3. C/S 모델에서의 클라이언트 네트워크 프로그래밍하기	
3-1. TCP 소켓 통신 네트워크 모듈 설계	18
3-2. 네트워크 패킷 구조 재구축	23
• 교수 · 학습 방법	33
• 평가	34
학습 4. P2P 모델의 클라이언트 네트워크 프로그래밍하기	
4-1. UDP 소켓 통신 네트워크 모듈 기획 및 구축	36
• 교수 · 학습 방법	40
• 평가	41

학습 5. 게임 서버 응용 프로그래밍하기

5-1. 게임 서버 구조 설계	43
5-2. 게임 서버 테스트 환경 구축	51
• 교수 · 학습 방법	54
• 평가	55

학습 6. 플레이(캐릭터) 프로그래밍하기

6-1. 입출력 데이터에 따른 캐릭터 반응 설계 및 구현	57
6-2. 플레이어 캐릭터 행동 구현	62
• 교수 · 학습 방법	68
• 평가	69

학습 7. 지형 프로그래밍하기

7-1. 지형과 월드 설계 및 구현	71
7-2. 지형 내비게이션 알고리즘 구현	74
• 교수 · 학습 방법	79
• 평가	80

학습 8. 이벤트 프로그래밍하기

8-1. 게임 플레이 이벤트 설계 및 구현	82
8-2. 상호작용 이벤트 프로그래밍 구현	86
• 교수 · 학습 방법	89
• 평가	90

참고 자료	92
-------------	----

게임 네트워크와 클라이언트 프로그래밍 학습모듈의 개요

학습모듈의 목표

온라인 게임뿐만 아니라 플랫폼 독립적으로 유무선 네트워크를 통하는 게임의 서버 컴퓨터에서, 게임이 원활하게 운영되도록 게임 서버 시스템을 설계하고, 네트워크 프로그램으로 구현하고, 게임 서버 응용 프로그램을 구현할 수 있다. 또한, 설정된 게임 세계의 배경 위에서 서로 상호작용하는 캐릭터와 오브젝트들의 규칙과 이벤트를 구현할 수 있다.

선수학습

게임 알고리즘과 게임엔진 프로그래밍(0803020526_18v4), 게임 데이터베이스와 웹 프로그래밍(0803020532_18v4)

학습모듈의 내용체계

학습	학습 내용	NCS 능력단위 요소	
		코드번호	요소 명칭
1. 게임 서버 시스템 설계하기	1-1. 게임 서버 시스템 파악 1-2. 게임 서버 시스템 설계	0803020534_18v4.1	게임 서버 시스템 설계하기
2. 서버 네트워크 프로그래밍하기	2-1. 네트워크 프로그래밍 제작	0803020534_18v4.2	서버 네트워크 프로그래밍하기
3. C/S 모델에서의 클라이언트 네트워크 프로그래밍하기	3-1. TCP 소켓 통신 네트워크 모듈 설계 3-2. 네트워크 패킷 구조 재구축	0803020534_18v4.3	C/S 모델에서의 클라이언트 네트워크 프로그래밍하기
4. P2P 모델의 클라이언트 네트워크 프로그래밍하기	4-1. UDP 소켓 통신 네트워크 모듈 기획 및 구축	0803020534_18v4.4	P2P 모델의 클라이언트 네트워크 프로그래밍하기
5. 게임 서버 응용 프로그래밍하기	5-1. 게임 서버 구조 설계 5-2. 게임 서버 테스트 환경 구축	0803020534_18v4.5	게임 서버 응용 프로그래밍하기
6. 플레이(캐릭터) 프로그래밍하기	6-1. 입출력 데이터에 따른 캐릭터 반응 설계 및 구현 6-2. 플레이어 캐릭터 행동 구현	0803020542_18v1.1	플레이(캐릭터) 프로그래밍하기
7. 지형 프로그래밍하기	7-1. 지형과 월드 설계 및 구현 7-2. 지형 내비게이션 알고리즘 구현	0803020542_18v1.2	지형 프로그래밍하기
8. 이벤트 프로그래밍하기	8-1. 게임 플레이 이벤트 설계 및 구현 8-2. 상호작용 이벤트 프로그래밍 구현	0803020542_18v1.3	이벤트 프로그래밍하기

핵심 용어

패킷 설계, 소켓, 프로토콜, 트랜잭션, 패치 서버, 인증 서버, 로비 서버, 채팅 서버, 분산 게임 서버, 패킷 암호화

학습 1

게임 서버 시스템 설계하기

학습 2	서버 네트워크 프로그래밍하기
학습 3	C/S 모델에서의 클라이언트 네트워크 프로그래밍하기
학습 4	P2P 모델의 클라이언트 네트워크 프로그래밍하기
학습 5	게임 서버 응용 프로그래밍하기
학습 6	플레이(캐릭터) 프로그래밍하기
학습 7	지형 프로그래밍하기
학습 8	이벤트 프로그래밍하기

1-1. 게임 서버 시스템 파악

학습 목표

- 게임 기획을 바탕으로 유무선 네트워크로 연동하는 게임의 서버 시스템을 구상할 수 있다.
- 서버 시스템 구현에 필요한 정보들을 수집, 분석할 수 있다.

필요 지식 /

① 클라이언트

컴퓨터 네트워크(computer network)를 통해 다른 컴퓨터에 접속하여 원격 서비스를 받는 종말 응용 프로그램이나 서비스(service)를 말한다. 글자 그대로 클라이언트(client)는 어떤 행위나 요청을 서버에 보내어 그 서버가 제작된 방식에 맞게 정보를 클라이언트에 제공한다. 우리가 보통 사용하는 스마트폰, PC 등을 클라이언트라고 부를 수도 있고, 그 기기에 탑재되어 있는 응용 프로그램이나 서비스를 말하기도 한다.

② 서버

서버(server)는 통신을 통해 클라이언트에게 필요한 데이터를 제공해서 애플리케이션(application)을 활용할 수 있도록 하는 컴퓨터 또는 프로그램을 지칭하며, 용도에 따라 파일 서버, 웹 서버, 메일 서버로도 분리하기도 하지만 최근에는 특수 목적의 서비스를 제공하기도 한다. 서버는 영문 명칭 그대로 '서비스를 제공하는 자(者)'이지만 특정하게 제작된 클라이언트를 위한 정보 제공 용도로도 많이 사용한다. IDC에 서버를 설치하여 서비스하기도 하지만 요즘에는 클라우드 서비스 업체에서 가상 서버를 임대해서 사용하기도 한다.

③ 다운로드/패치 서버

과거에는 클라이언트를 사용자에게 제공하기 위해서는 CD와 같은 미디어를 이용하여 배포하였지만 배포하는 데 오래 걸린다는 단점이 있었다. 클라이언트 소프트웨어를 사용자가 집이나 회사, 이동 중에도 쉽게 클릭만으로 다운로드받아 설치할 수 있도록 CDN 서비스를 이용하여 다운로드 서버를 구축하는 편이다. 패치 서버는 게임의 버전을 관리하여 클라이언트의 버전을 최신으로 유지하는 역할을 한다. 게임 서버나 클라이언트는 업데이트를 해야 할 필요가 있어 클라이언트가 자신의 설치된 프로그램을 최신의 상태로 만들 수 있는 서버를 두어 자동으로 업데이트 데이터를 받을 수 있게 한다.

④ 채널 서버

규모가 큰 서버를 제작할 경우 게임 내의 사용자 혼잡도에 따라 서버를 여러 대 두어야 하는 경우가 발생하기도 한다. 이때, 사용자가 정확하게 자신이 만든 서버로 갈 수 있도록 도와주는 서버이다. 간혹 로비 서버가 채널 서버의 기능을 맡아서 하거나 게임 서버 내에서 다른 채널로 이동하기 위해 채널 서버 기능을 게임 서버에 두기도 한다.

수행 내용 / 게임 서버 시스템 파악하기

재료·자료

- 게임 기획서, 종이, 필기도구

기기(장비 · 공구)

- 화이트보드, 컴퓨터 등

안전 · 유의 사항

- 해당 사항 없음.

수행 순서

① 네트워크 게임을 조사하고 형태를 분류한다.

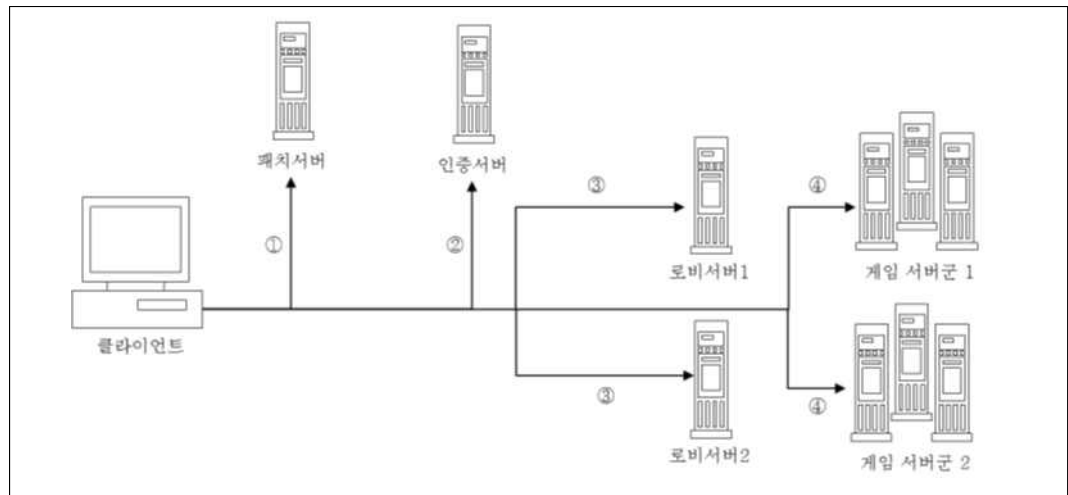
1. 다양한 형태의 네트워크 게임을 조사한다.

- (1) 게임의 장르와 상관없이 네트워크를 이용하여 즐기는 다양한 게임 정보를 수집한다.
- (2) 게임의 특징과 네트워크 게임 플레이의 차이점이 존재하는 게임 위주로 게임 정보를 수집한다.
- (3) 게임의 특징과 장르, 또는 유·무선 네트워크에 따라 게임 품질에 어떤 영향을 보이는지 체크한다.

2. 조사된 네트워크 게임을 중심으로 네트워크 플레이 형태를 분류하고, 각 플레이 형태별 특징을 조사한다.

예) MMPOG(massive multi player online game), 웹(web game), 모바일 게임(mobile game) 등

3. 네트워크 플레이 형태별로 분류된 게임 목록에서 각 항목별 대표적인 게임을 선정해 뒤, 선정된 게임 내용을 분석하여 네트워크 시스템의 구성을 유추하고 네트워크 시스템 구성도를 작성한다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력 개발원

[그림 1-1] 서버군의 역할 설정과 접속 순서 예시

② 작성된 네트워크 시스템을 분석한다.

시스템 구조의 특징과 서버 시스템 구성원의 역할, 각 네트워크 플레이 형태별 시스템 구조의 차이점을 비교한다.

③ 비교한 네트워크 시스템의 구성을 토대로 서버 구축에 필요한 자료를 수집한다.

서버 구축에 필요한 기본 지식은 다양하게 있으며 쓰임새에 따라 서버 구조가 달라질 수 있으므로 필요한 자료를 충분히 조사한다.

수행 tip

- 네트워크 게임을 조사할 때에는 게임에 적용된 네트워크의 특성과 연동 구성이 서로 다른 다양한 게임을 조사한다.
- 게임 클라이언트의 유·무선 접속에 따른 네트워크 응답 속도, 회선 단절과 재접속에 대한 대응 방법을 다양한 게임을 조사하여 분석한다.

1-2. 게임 서버 시스템 설계

학습 목표

- 분석한 내용에 따라 서버 시스템 구조를 설계할 수 있다.

필요 지식 /

① 블록 다이어그램(block diagram)

데이터 또는 정보의 흐름과 관련한 자료 분석을 시작할 때, 자료의 처리 과정을 단계적으로 설계하여 하나의 그림으로 나타낸 것을 뜻한다. 이때, 각 자료와 단계는 블록으로 처리되며, 단계의 흐름은 선으로 표현되어 각각의 자료와 단계를 연결하는 형식으로 구성된다. 보통 하나의 단계가 블록으로 표시되어 블록 다이어그램이라고 부른다. 다른 말로 블록 차트(block chart), 순서도(flow chart)라고도 한다.

② 게임 서버

게임 서버는 사용자가 게임에 들어가 실제 게임을 진행하는 동안에 접속하여 여러 기능을 제공하는 서버를 말한다. 사용자가 선택한 캐릭터로 게임을 즐길 수 있는 공간을 마련해 주는 서버로, 가장 복잡하지만 최고의 성능과 견고함으로 무장해야 한다. 데이터 해킹에 가장 많이 신경 써야 하기도 하지만, 무엇보다 클라이언트/서버 간의 통신 시 TCP/IP, 메모리 취약점을 가장 많이 신경 써야 한다.

③ 분산 게임 서버

단일 서버에서 수용할 수 있는 게임 사용자의 수에는 한계가 있다. 따라서 좀 더 많은 사용자가 하나의 게임 월드 내에서 자유롭게 게임을 즐기게 하기 위한 대안으로 여러 대의 서버를 하나로 묶거나 사용자가 필요 시 기능별로 분산 접속할 수 있도록 만든 서버를 분산 게임 서버라고 한다.

수행 내용 / 게임 서버 시스템 설계하기

재료·자료

- 게임 기획서, 종이, 필기도구

기기(장비·공구)

- 화이트보드, 컴퓨터 등

안전·유의 사항

- 해당 사항 없음.

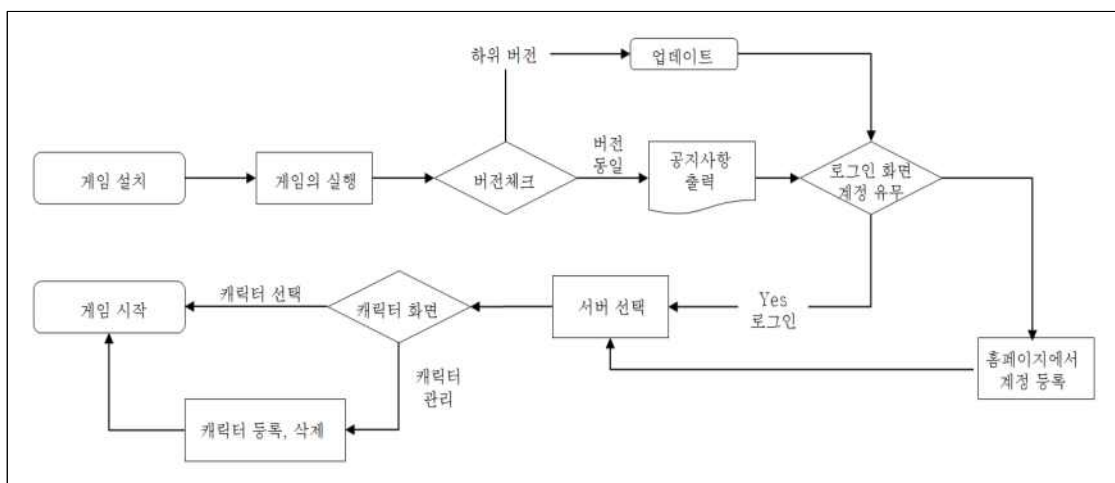
수행 순서

① 게임 서버 시스템을 설계한다.

1. 각 팀별로 준비된 게임 기획서를 분석하고, 조사한 다양한 네트워크 시스템 중 가장 적합한 네트워크 모델을 설정한다.

게임 장르와 플레이 형태를 중심으로 게임 기획서를 분석하여 이미 조사된 다양한 게임 장르와 유사한 형태의 네트워크 모델을 설정한다.

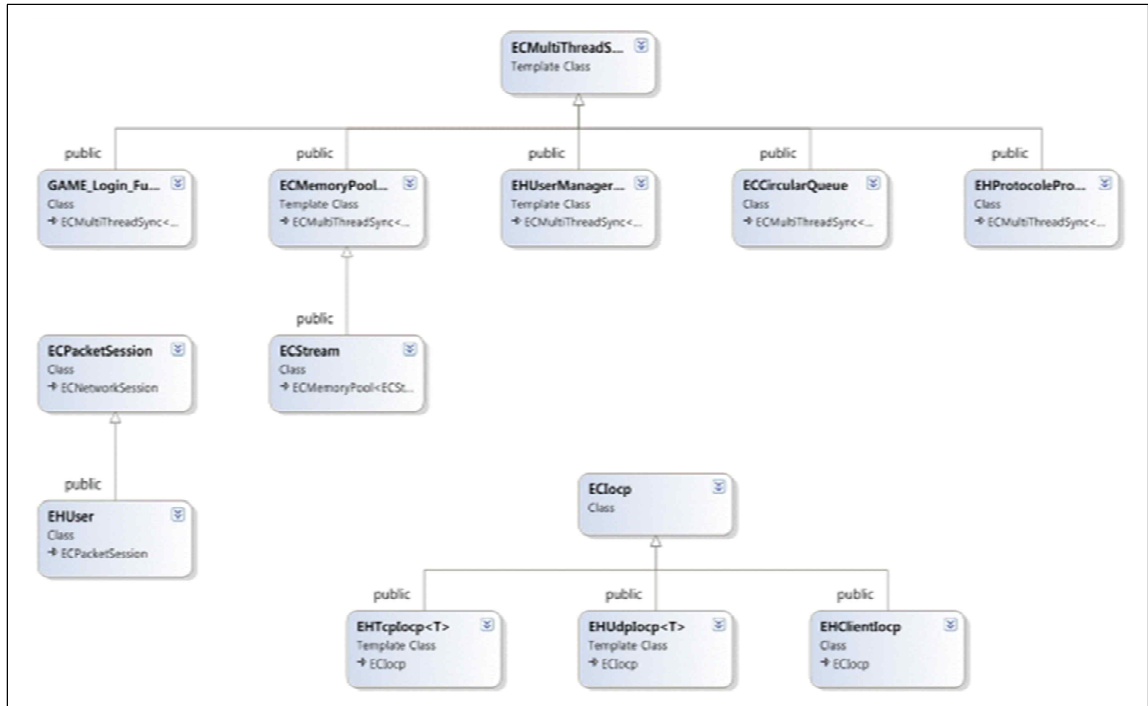
2. 게임 기획서의 분석 내용을 바탕으로 네트워크상에 송수신되어야 할 게임 데이터를 추출하고 분석된 데이터로부터 서버군의 데이터 흐름도를 작성한다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원.
[그림 1-2] 서버의 데이터 흐름도 예시

3. 서버군의 데이터 흐름도를 중심으로 서버군을 구성하고, 서버군 내 서버 각각의 역할을 설정한다.
4. 서버와 클라이언트 간의 데이터 통신을 위한 프로토콜(protocol)을 설계하고, 데이터 전송 패킷(packet)을 설계한다.

5. 역할이 설정된 서버와 클라이언트가 작성해야 할 프로그램 모듈을 설계한다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원 [그림 1-3] 서버 프로그램 모듈 설계의 예

② 프로그램 모듈별 기능과 입출력 데이터를 정의한 프로그램 명세표를 작성한다.

〈표 1-1〉 서버 프로그램 명세표의 예

모듈 ID	GAME_Login_Func
목적	게임 사용자 계정의 유효성을 판정하고, 유효한 사용자일 경우 계정에 필요한 게임 계정 정보를 데이터베이스 모듈에 요청하고 요청된 데이터를 수신하는 역할을 수행
입력 데이터	사용자 ID(char[64]), 비밀번호(char[32]) 조건 : 사용자 인증 성공 유무
출력 데이터	인증 성공 : Msg_OK 반환 인증 실패 : Msg_Fail 반환
사용방법	첫 번째 인자 값으로 사용자 ID 문자열, 두 번째 문자열로 패스워드 문자열 설정 후 함수 호출
효과	[인증 성공] 해당 아이디의 완성된 사용자 계정 정보를 데이터베이스에서 가져와 해당 사용자 계정에 데이터 설정 [인증 실패] 사용자 정보의 인증 실패로 에러 메시지를 발생시킴 사용자 ID 검색 실패 : 해당 아이디 없음 메시지 출력
예외사항	(발생 메시지 : CERT_ERR_ID) 비밀번호 일치 실패 : 비밀 번호 오류 메시지 출력 (발생 메시지 : CERT_ERR_PWD) 데이터베이스의 검색 권한 요청 실패시 오류 메시지 출력 (발생 메시지 : DB_ERR_CNT)
오류	CERT_ERR_ID, CERT_ERR_PWD, DB_ERR_CNT

교수 방법

- 데이터 통신의 이론적인 지식을 충분히 전달하고 이해시킨 다음 단계적인 실습을 지도한다.
- 게임 서버 시스템 분석은 서버군을 구성하고 있는 다양한 서버의 역할 및 서버와 서버 간, 그리고 서버와 클라이언트 간의 데이터 흐름을 중심으로 지도한다.
- 게임 클라이언트의 유·무선 접속에 따라 게임의 개발 방향 및 구조가 어떤 식으로 변경되는지 이해시키고 서버 설계에 반영할 수 있도록 지도한다.
- 게임 서버 시스템 구축 학습은 기초적인 서버와 클라이언트 간의 통신 흐름을 중심으로 지도한다.

학습 방법

- 데이터 통신에 관련된 용어를 학습한다.
- 게임 클라이언트의 유·무선 접속 네트워크의 차이점과 서버 설계 및 구조의 차이점을 이해한다.
- 게임 서버군의 구성과 각각의 역할 및 필요성을 이해한다.
- 게임 서버 간의 데이터 흐름과 그 역할을 이해한다.

학습 1 평 가

평가 준거

- 평가자는 학습자가 학습 목표를 성공적으로 달성하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가해야 한다.

학습 내용	학습 목표	성취수준		
		상	중	하
게임 서버 시스템 파악	- 게임 기획을 바탕으로 유무선 네트워크로 연동하는 게임의 서버 시스템을 구상할 수 있다.			
	- 서버 시스템 구현에 필요한 정보들을 수집, 분석할 수 있다.			
게임 서버 시스템 설계	- 분석한 내용에 따라 서버 시스템 구조를 설계할 수 있다.			

평가 방법

- 포트폴리오

학습 내용	평가 항목	성취수준		
		상	중	하
게임 서버 시스템 파악	- 게임 서버 구성을 이해하고 있는지 여부			
	- 게임 서버 각 구성 요소의 역할을 이해하고 있는지 여부			
게임 서버 시스템 설계	- 지정한 게임에 대한 게임 서버 분석 능력			
	- 주어진 게임 기획서를 바탕으로 구성한 게임 서버 설계 능력			

- 평가자 체크리스트

학습 내용	평가 항목	성취수준		
		상	중	하
게임 서버 시스템 파악	- 게임 서버 구성에 대한 이해 정도			
	- 게임 서버 각 구성 요소의 역할에 대한 이해 정도			
게임 서버 시스템 설계	- 게임 서버군 내 서버별 역할 분담에 대한 이해 정도			
	- 주어진 기획서를 바탕으로 게임 서버 설계 능력			

피드백

1. 포트폴리오

- 학습자가 제출한 서버 분석 내용과 서버 구성도가 기획 의도에 맞게 설계되었는지에 대해 설명해 준다.
- 주어진 게임 기획서를 토대로 작성한 게임 서버 설계 문서가 각 구성에 맞게 서로 유기적으로 작동하는지에 대해 설명해 준다.
- 서버 구조에 따른 역할 분담을 충분히 이해하고 설계되었는지에 대해 설명해 준다.
- 평가 결과, 게임 기획서에 적합한 서버 구조를 설계한 학습자들에게는 구조를 통한 데이터 전달에 대해 설명해 준다.
- 평가 결과, 게임 서버 시스템에 대해서 충분히 파악한 학습자들에게는 설계를 할 수 있도록 지도해 준다.

2. 평가자 체크리스트

- 게임 서버군 구성 간의 이해를 적절한 예시를 통해 체크한다.
- 게임 서버군 내 서버별 역할 분담을 예시를 통해 체크한다.
- 게임 기획을 바탕으로 만든 서버 설계 문서가 동작에 유효한지에 대해 주요 사항을 표시하여 체크한다.

학습 1	게임 서버 시스템 설계하기
학습 2	서버 네트워크 프로그래밍하기
학습 3	C/S 모델에서의 클라이언트 네트워크 프로그래밍하기
학습 4	P2P 모델의 클라이언트 네트워크 프로그래밍하기
학습 5	게임 서버 응용 프로그래밍하기
학습 6	플레이(캐릭터) 프로그래밍하기
학습 7	지형 프로그래밍하기
학습 8	이벤트 프로그래밍하기

2-1. 네트워크 프로그래밍 제작

학습 목표

- 데이터 통신의 원리와 기초를 알고 프로세스 간 통신을 구현할 수 있다.
- TCP/IP 통신을 구현할 수 있다.
- UDP 통신을 구현할 수 있다.

필요 지식 /

① 게임 서버 환경 구축

1. 리눅스(Linux)

리눅스는 리누스 토르발스가 커뮤니티 주체로 개발한 컴퓨터 핵심 운영 체제이다. 리눅스는 자유 소프트웨어와 오픈 소스 개발의 가장 유명한 표본으로 들 수 있다. 리눅스는 다중 사용자, 다중 작업(멀티태스킹; (multitasking)), 다중 스레드(thread)를 지원하는 네트워크 운영 체제(NOS)이다.

2. 윈도우(Windows)

마이크로소프트사가 개발한 컴퓨터 운영 체제이다. 애초에 개인용 컴퓨터에 처음으로 도입한 그래픽 사용자 인터페이스(GUI) 운영 체제인 맥 OS에 대항하여, 당시 널리 쓰이던 MS-DOS에서 멀티태스킹과 GUI 환경을 제공하기 위한 응용 프로그램으로 처음 출시되었다. 현재 전 세계 90%의 개인용 컴퓨터에서 쓰고 있으며, 서버용 운영체제로도 점차 영역을 넓혀 나가고 있다. 윈도우 운영체제의 경우 큰 시장 점유율을 차지하고 있는 까닭에 일반 사용자들에게 매우 익숙할 뿐 아니라 호환되는 유명한 응용 프로그램이 많다는 장점을 지니고 있다. 따라서 그만큼 보안 문제에서는 취약한 부분이 많

은 운영체제로 인식되기도 하지만 사실 보안 문제의 취약성이 다른 운영체제와 크게 차이 날 정도로 심각한 것은 아니다.

3. 웹 서버(web server)

웹 서버는 HTTP를 통해 웹 브라우저에서 요청하는 HTML 문서나 오브젝트(이미지 파일 등)를 전송하는 서비스 프로그램을 말한다. 웹 서버 소프트웨어를 구동하는 하드웨어도 웹 서버로 혼동하는 경우가 가끔 있다.

4. 데이터베이스(database)

데이터베이스는 여러 사람이 공유하고 사용할 목적으로 통합 관리되는 정보의 집합이다. 논리적으로 연관된 하나 이상의 자료의 모음으로, 그 내용을 고도로 구조화함으로써 검색과 갱신의 효율화를 꾀하였다. 즉, 몇 개의 자료 파일을 조직적으로 통합하여 자료 항목의 중복을 없애고 자료를 구조화하여 기억시켜 놓은 자료의 집합체이다.

수행 내용 / 네트워크 프로그래밍 제작하기

재료·자료

- 프로토콜 설계 문서, 전송 패킷 구조도, 온라인 게임 서버 구조도, 게임 기획서, 소켓 프로그래밍 참고서, IOCP 관련 참고서

기기(장비 · 공구)

- 컴퓨터, VOOOOO SOOOOO 2019 이상 버전, MyOOO 데이터베이스 설치 프로그램

안전 · 유의 사항

- 2인 1조로 팀을 구성하여 클라이언트, 서버의 역할을 수행한다.
- 코드 구현과 실행 시 프로그램이 비정상적으로 종료될 때가 있으므로 항상 저장과 관리에 유의한다.

수행 순서

① 게임 서버 환경을 구축한다.

1. 리눅스 또는 윈도우 서버 운영체제를 설치한다.

향후 소켓 프로그래밍과 서버 프로그래밍을 고려하여 적합한 환경의 OS를 설치한다. 윈도우 10 이상의 운영체제를 사용하는 PC의 경우, 운영체제의 재설치 없이 기존의 운영체제로도 서버 역할을 수행할 수 있다.



스크린 샷은 Microsoft Corporation으로부터 허가를 받아 사용하였습니다.
출처: Microsoft(<https://www.microsoft.com>)에서 2022. 9. 26. 스크린샷.
[그림 2-1] Windows Server



출처: U사(<https://ubuntu.com>)에서 2022. 9. 26. 스크린샷.
[그림 2-2] LiOOO OS 중 하나인 UbOOOO

2. 웹 서버(web server) 프로그램을 설치한다.(예: IIS, Apache 등)
간단한 HTML 문서를 작성하고, 웹 페이지가 정상적으로 출력되는지 확인한다.
3. 데이터베이스 프로그램을 설치한다.
데이터베이스 프로그램을 설치한 뒤, 간단한 테이블 생성 및 SQL문을 실행하여 정상 설치 여부를 확인한다.
4. 프로그래밍에 필요한 SDK와 개발 도구를 설치한다.
서버 프로그래밍을 위하여 자바(JAVA)와 자바 프로그래밍 도구 또는 기타 개발 도구와 개발 환경에 적합한 프로그램을 설치하고, 간단한 프로그램을 작성하고 실행하여 정상 설치 여부를 확인한다.

수행 tip

- 게임 서버 구축 실습 시에는 서버를 구성하기 위한 필요 구성 요소를 학습하고, 기본 구성 요소에 대한 기본적인 운영 환경을 설정할 수 있는 능력을 배양하는 것을 주된 목표로 한다.

교수 방법

- 데이터 통신의 이론적인 지식을 충분히 전달하고 이해시킨 다음 단계적인 실습을 지도한다.
- 네트워크 프로토콜 구현 학습은 기초적인 서버와 클라이언트 간 데이터 전송과 수신에 대한 이해, 그리고 서버와 클라이언트 간 소켓 연결과 연결 상태 유지를 중심으로 지도한다.
- 실습은 2인 1조 또는 그 이상의 팀으로 구성하여 서버와 클라이언트의 역할을 각각 담당시킨 뒤에 실습 지도를 수행한다.
- 실습 중 서버와 클라이언트의 구현 작업은 팀과 상관없이 학습자 개인별로 서버 프로그램과 클라이언트 프로그램을 모두 구현하되, 서버 프로그램과 클라이언트 프로그램의 결과 확인은 2인 1조로 수행하도록 지도한다.

학습 방법

- 데이터 통신에 관련된 용어를 이해하고, 서버와 클라이언트 간의 데이터 전송과 수신 체계를 이해하는 데 중점을 두어 학습한다.
- 게임 서버군의 각 서버 간 전송 데이터와 수신 데이터의 흐름을 중심으로 게임 서버의 역할을 이해한다.
- 데이터 통신의 주된 프로토콜인 TCP를 이해하고, TCP에서 지원하는 다양한 소켓 옵션을 이해한다.

평가 준거

- 평가자는 학습자가 학습 목표를 성공적으로 달성하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가해야 한다.

학습 내용	학습 목표	성취수준		
		상	중	하
네트워크 프로그래밍 제작	- 데이터 통신의 원리와 기초를 알고 프로세스 간 통신을 구현할 수 있다.			
	- TCP/IP 통신을 구현할 수 있다.			
	- UDP 통신을 구현할 수 있다.			

평가 방법

- 포트폴리오

학습 내용	평가 항목	성취수준		
		상	중	하
네트워크 프로그래밍 제작	- 서버 프로그램 구축 능력			
	- 클라이언트 프로그램 구현 능력			
	- 프로그램 개발 문서 작성 능력			

- 서술형 시험

학습 내용	평가 항목	성취수준		
		상	중	하
네트워크 프로그래밍 제작	- 서버 프로그램의 순차적인 절차와 기능을 명확하게 숙지하였는지 여부			
	- 클라이언트 프로그램의 순차적인 절차와 기능을 명확하게 숙지하였는지 여부			
	- 개발 프로그램의 문서화가 왜 필요한지에 대해 이해하고 있는지 여부			

• 평가자 체크리스트

학습 내용	평가 항목	성취수준		
		상	중	하
네트워크 프로그래밍 제작	- TCP 프로토콜에 대해 이해하고 있는지 여부			
	- 서버 프로그램과 클라이언트 프로그램 간의 통신을 위한 순차적 역할에 대해 이해하고 있는지 여부			
	- 개발 프로그램의 문서화 능력			

피드백

1. 포트폴리오

- 서버 프로그램에 대한 모범적인 서버 프로그램의 소스 코드를 제시하고, 문법적 · 논리적 오류를 설명해준다.
- 클라이언트 프로그램에 대한 모범적인 서버 프로그램의 소스 코드를 제시하고, 문법적 · 논리적 오류를 설명해준다.
- 프로그램 개발 문서 검토 후 문서화의 오류 및 누락, 잘못된 내용을 분석하여 설명해준다.

2. 서술형 시험

- 개발 프로그램에 대한 문서화의 필요성과 역할에 대하여 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 다시 설명해준다.
- 개발 프로그램에 대한 문서화의 완성도에 대해서 평가한 뒤, 완성도가 우수한 학습자에 대해서는 응용할 수 있는 부분을 설명해준다.
- 평가 결과, TCP 통신에 대해서 충분히 이해한 학습자들에게는 UDP 통신에 대해서 설명해 준다.

3. 평가자 체크리스트

- TCP/IP 프로토콜을 정확히 이해할 수 있도록 표와 블록다이어그램으로 체크한다.
- 서버/클라이언트 간의 통신 정의 및 역할, 순서에 대하여 체크한다.
- 개발 프로그램 문서가 동작 흐름에 맞게 충분히 설명되었는지 체크한다.

학습 1	게임 서버 시스템 설계하기
학습 2	서버 네트워크 프로그래밍하기
학습 3	C/S 모델에서의 클라이언트 네트워크 프로그래밍하기
학습 4	P2P 모델의 클라이언트 네트워크 프로그래밍하기
학습 5	게임 서버 응용 프로그래밍하기
학습 6	플레이(캐릭터) 프로그래밍하기
학습 7	지형 프로그래밍하기
학습 8	이벤트 프로그래밍하기

3-1. TCP 소켓 통신 네트워크 모듈 설계

학습 목표

- TCP/IP 소켓 통신을 활용한 네트워크 모듈을 제작할 수 있다.

필요 지식 /

① 게임 프로토콜 설계

1. 프로토콜

(1) 정의

네트워크에서의 프로토콜은 두 개체(컴퓨터와 단말기, 컴퓨터나 컴퓨터 같은 다수의 통신 주체로 데이터를 전송하는 자와 데이터를 수신하는 자로 구분) 간의 통신 규약을 의미한다. 두 개체가 효율적인 상호 연동을 위해 상호간에 요청하거나 명령하는 규칙을 약속한 통신 규약을 프로토콜이라고 한다. 통신 규약은 상호간의 접속이나 전달 방식, 통신 방식, 주고받을 자료의 형식, 오류 검출 방식, 코드 변환 방식, 전송 속도 등에 대하여 정하는 것을 말한다.

(2) 목적

프로토콜은 그 목적에 따라 여러 가지 형태가 있다. 대표적인 프로토콜의 예로 컴퓨터 통신에 범용적으로 사용되고 있는 TCP(transmission control protocol)와 UDP(user datagram protocol)가 있다. 일반적으로 통신 규약이 필요한 두 개체 사이에는 상호 간 거리가 존재한다. 따라서 정보를 전송하기 위해서는 정보를 전기적인 신호의 형태로

변환하고 그 변환된 신호가 통신망을 통해 흐르도록 하는데, 통신망에는 정상적인 신호의 흐름을 방해하는 여러 가지 현상이 존재하게 된다. 이러한 현상은 정확한 정보의 전송을 방해하여 전기적 신호의 흐름을 방해하거나 유실, 변형이 발생하여 정보 전송의 오류가 발생하는 원인이 된다. 프로토콜이라는 규약의 집합 속에는 이러한 오류에 대응하기 위한 약속이 대단히 중요하다.

(3) 특징

게임 서버와 클라이언트 사이에도 프로토콜의 개념이 필요하다. 게임 정보의 전송과 수신을 위해서 TCP 또는 UDP 등을 이용하겠지만, 이외에도 게임 서버 프로그램과 클라이언트 게임 프로그램 사이에서 게임을 연동시키기 위해 어떤 행위를 요청하거나 명령하기 위한 세부적인 통신 규칙을 정한다. 이와 같이, 사용자가 정의한 세부적인 통신 규칙도 프로토콜에 해당한다.

2. 패킷(packet)

패킷은 데이터 전송에서 송신측과 수신측에 의해 하나의 단위로 취급되어 전송되는 집합체를 의미한다. 즉, 많은 양의 전송 데이터를 수신측으로 이동시키기 위해서 일정 크기로 데이터를 분할하게 되는데, 이렇게 단위 크기로 분할된 하나의 데이터 덩어리를 패킷이라고 한다. 게임 서버와 클라이언트는 수많은 작업 요청과 명령이 오가면서 통신을 한다. 게임의 실시간적 연동 특성을 표현하기 위해서는 1초당 수십 번에서 많게는 수백 번의 데이터의 전송과 수신이 빠르게 이루어져야 한다. 또한, 네트워크에는 대역폭이라는 데이터 전송의 한계가 존재한다. 따라서 패킷의 구조 설계는 패킷의 정보를 최소화시키고 패킷의 크기를 최소화시키는 것에 중점을 두어 진행해야 한다.

3. TCP/UDP 프로토콜

(1) TCP(transmission control protocol)

전송 제어 프로토콜(TCP)은 인터넷 프로토콜 스위트(IP)의 핵심 프로토콜 중 하나로, IP와 함께 TCP/IP라는 명칭으로도 널리 불린다. TCP는 근거리 통신망이나 인트라넷, 인터넷에 연결된 컴퓨터에서 실행되는 프로그램 간에 일련의 옥텟(octet)을 안정적으로, 순서대로, 그리고 에러 없이 교환할 수 있게 한다. TCP는 전송 계층에 위치한다. 네트워크의 정보 전달을 통제하는 프로토콜이자 인터넷을 이루는 핵심 프로토콜의 하나로서 국제 인터넷 표준화 기구(IETF)의 'RFC 793'에 기술되어 있다.

수행 내용 / TCP 소켓 통신 네트워크 모듈 설계하기

재료·자료

- 프로토콜 설계 문서, 전송 패킷 구조도, 온라인 게임 서버 구조도, 게임 기획서, 소켓 프로그래밍 참고서

기기(장비·공구)

- 컴퓨터, VOOOOO SOOOOO 2019 이상 버전, MyOOO 데이터베이스 설치 프로그램

안전·유의 사항

- 2인 1조로 팀을 구성하여 클라이언트, 서버의 역할을 수행한다.
- 코드 구현과 실행 시 프로그램의 비정상 종료가 예상되므로 저장과 관리에 항상 유의한다.

수행 순서

① 게임 프로토콜을 설계한다.

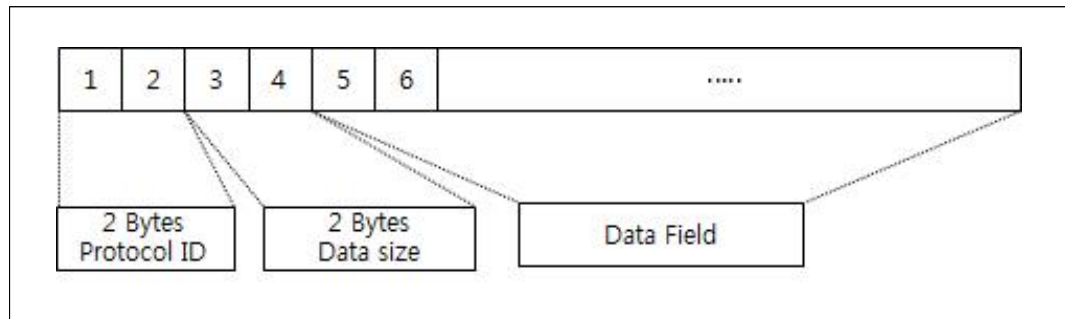
1. 게임 기획서를 분석하여 게임 서버와 클라이언트 간에 전송되어야 할 프로토콜을 정한다.
 - (1) 안전하게 통신해야 할 정보는 TCP 프로토콜을 사용하면 정보를 주고받으면서 데이터 유실을 방지할 수 있다.
 - (2) 데이터가 유실되더라도 고속의 정보를 주고받고 싶을 경우에는 UDP 프로토콜을 사용하여 데이터를 주고받을 수 있다.
UDP 프로토콜을 사용하더라도 안전하게 정보를 주고받을 수 있게 만들 수 있지만, 프로그래머 능력에 좌우되기 때문에 유실에 민감하지 않은 정보를 주로 사용한다.
2. 정의된 프로토콜에 따라 전송되어야 할 게임의 상태 값을 분류하고, 각 상태별 주요 데이터 목록을 작성한다.

〈표 3-1〉 게임 상태별 주요 전송 데이터 목록 예시

레벨	게임 상태와 주요 변수	상태 설명	자료형
1	로그인	게임 실행 시 로그인 화면 상태	
1-1	로그인 아이디	사용자의 아이디 값	char[32]
1-2	로그인 비밀번호	사용자의 비밀번호 값	char[32]
1-3	로그인 결괏값	사용자 로그인 결과 메시지 값	char
2	대기실	게임의 대기실 입장 상태	
2-1	게임 캐릭터명	게임 내 캐릭터 이름	char[32]
2-2	게임 캐릭터 레벨	게임 내 캐릭터의 레벨 값	int
⋮	⋮	⋮	⋮

3. 게임 기획서에 맞게 패킷 클래스를 설계한다.

팀별로 주어진 게임 기획서를 토대로 기획서의 기획 특성에 맞게 전송 패킷 클래스를 설계한다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력 개발원

[그림 3-1] 패킷 클래스(packet class) 구조 설계 예시

4. 게임 상태별 주요 전송 데이터 목록과 패킷 클래스의 구조를 토대로 게임 내에서 서버와 클라이언트가 서로 주고받을 프로토콜을 설계한다.

팀별로 주어진 게임 기획서에 맞게 서버와 클라이언트가 주고받을 통신 프로토콜을 설계하고, 목록을 작성한다.

〈표 3-2〉 게임 프로토콜 설계 문서 예시

프로토콜 이름	프로토콜 ID 값	데이터 크기	데이터 형식	전송 데이터
GS_ENTERROOM_REQ	101	없음.	없음.	접속한 사용자의 계정 정보를 요구한다.
GS_ENTERROOM_ACK	102	총128Byte		'GS_ENTERROOM_REQ'의 요청에 따른 사용자 계정의 정보를 응답 전송한다.
		1Bit	bool	계정 요청 작업의 성공 유무 - true: 계정 정보 접근 성공 - false: 계정 정보 접근 실패
		32Byte	String	게임 캐릭터 이름
		1Byte	int	게임 캐릭터의 레벨
		3Bit	int	게임 캐릭터의 타입
⋮		⋮	⋮	⋮
GS_ENTERGAME_REQ	201	1Byte	int	인자로 전송된 번호에 해당하는 서버로 진입을 요청한다.
⋮		⋮	⋮	⋮

4. 프로토콜 설계의 무결성을 검증하기 위해 각 팀 내에서 구성원들이 각각 서버와 클라이언트의 역할을 할당하고, 프로토콜 카드를 만들어 모의 클라이언트 서버 통신 시뮬레이션을 수행한다.
5. 모의 클라이언트 서버 통신 시뮬레이션 역할극을 통해 발견된 누락 프로토콜이나 잘못된 데이터를 수정하여 최종 프로토콜 설계 문서를 완성한다.

수행 tip

- 네트워크를 통한 통신이라는 한계를 고려하여 서버와 클라이언트 간의 전송과 수신 데이터는 최소의 크기로 설계한다.
- 모의 클라이언트 서버 통신 시뮬레이션 역할극은 설계상에서 누락되거나 중복 및 오류 정보를 발견하기 위한 것이므로, 교수자에 의해 다른 방법으로 검증 방법을 대체할 수 있다.

3-2. 네트워크 패킷 구조 재구축

학습 목표

- 게임 루프와 별개로 돌아가는 네트워크 스레드를 구현할 수 있다.
- 패킷 구조를 설계하고 암호화 기법을 활용하여 패킷의 보안을 강화할 수 있다.
- 패킷 부하 테스트를 통해 네트워크 모듈의 안정성을 검증하고 최적화할 수 있다.
- 패킷 구조와 크기를 서버 트래픽을 고려하여 구현할 수 있다.

필요 지식 /

① 서버 프로그램 구현

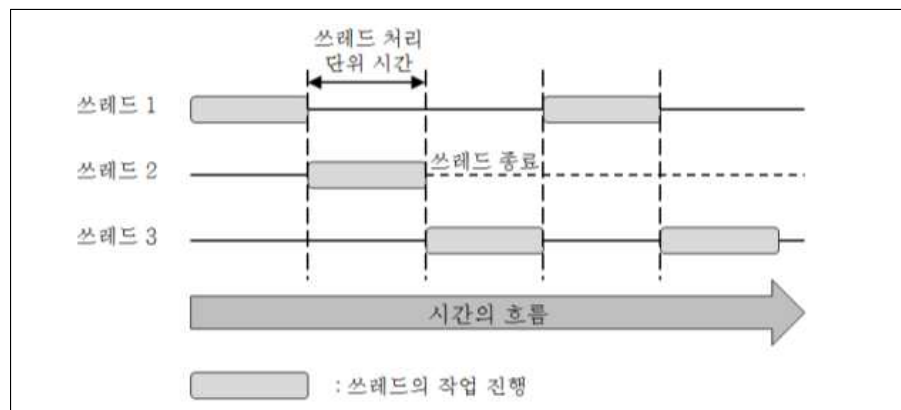
1. 스레드(thread)

(1) 정의

스레드란 하나의 작업 프로세스를 진행시키는 단위를 말한다. 프로그램 하나를 만들 때, 그 프로그램은 메인(main)을 갖고 있으며, 프로그램은 메인에서부터 시작한다. 운영 체제는 실행될 프로그램에 스레드를 하나 할당하고, 그 스레드는 메인 함수를 실행한다. 이렇게 실행된 프로그램에는 시스템으로부터 기본적으로 하나의 스레드가 주어지는데, 이를 메인 스레드(main thread)라고 한다.

(2) 목적

운영 체제는 스레드 하나가 한 번에 진행할 수 있는 단위를 정하고, 각 스레드들의 작업을 조금씩 진행시키는 병렬 처리를 수행한다. 이렇게 병렬로 작업되는 스레드를 응용 프로그램에서 임의로 생성하고 작업을 할당할 수 있다. 즉, 사용자가 작성하는 프로그램이 내부 작업을 병렬로 처리할 수 있다는 의미이다.



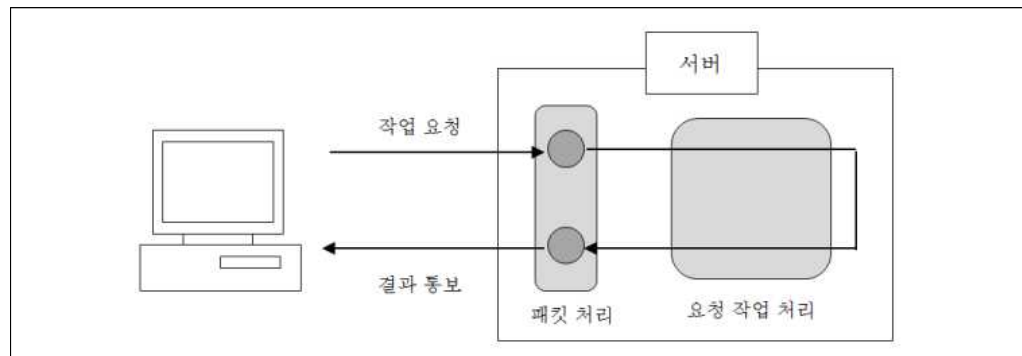
출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원

[그림 3-2] 스레드에 할당된 작업 시간

(3) 적용

하나의 서버에 2개의 클라이언트가 접속하였을 때, 클라이언트들은 서버에 작업 요청을 하게 된다. 그런데 만일 서버에 메인 스레드(thread)만으로 모든 작업을 처리하면 클라이언트로부터 들어오는 패킷을 쌓아놓고 순차적으로 읽어 들여 처리하고 다시 통보해 주어야 하기 때문에 클라이언트의 작업 요청 응답 시간은 늦어지게 된다.

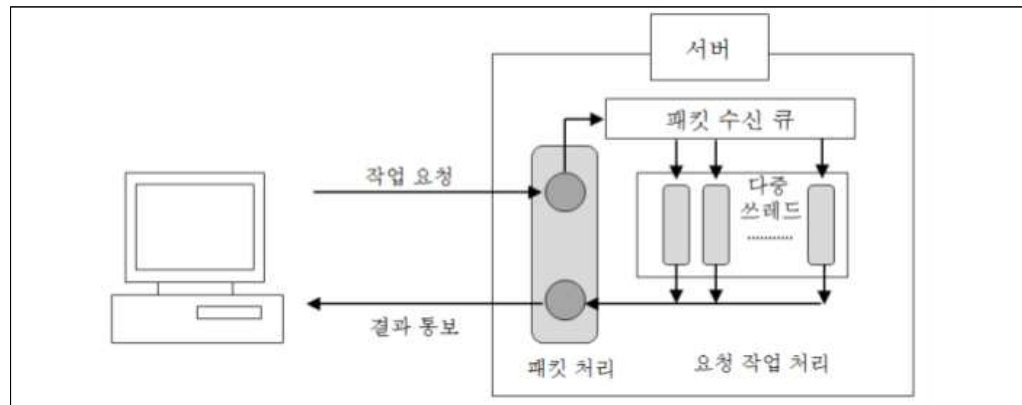
이와 같은 패킷의 병목 현상 때문에 게임 서버와 같은 다중 사용자의 접속을 동시에 처리하기 위해서는 다중 스레드 방식으로 처리해야 한다. 여러 개의 스레드를 생성하면 네트워크 작업과 내부 연산 작업을 병렬로 처리함으로써 처리 속도를 높일 수 있다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).

한국직업능력개발원

[그림 3-3] 서버의 단일 스레드 작업



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).

한국직업능력개발원

[그림 3-4] 다중 스레드를 이용한 병렬 처리

2. 문맥 교환(context switching)

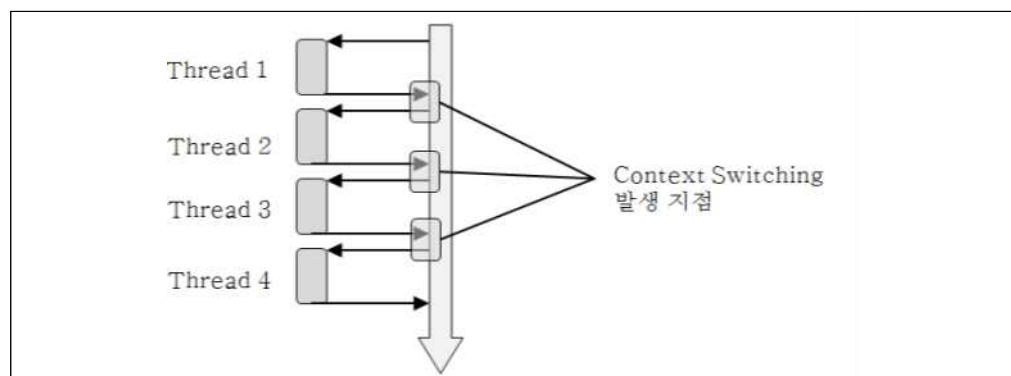
(1) 정의

운영체제는 스레드 하나의 작업을 진행하기 위해 그 스레드의 문맥(context)을 읽어오며, 다시 다른 스레드로 작업을 변경할 때 이전 스레드의 문맥을 저장하고 다시 진행할 스레드의 문맥을 읽어오는 작업을 반복한다. 이처럼 한 스레드에서 다른 스레드로 작업을 넘기는 과정을 문맥 교환(context switching)이라고 한다.

(2) 개념

응용 프로그램 차원에서 보면 문맥 교환 작업은 아주 작은 단위이다. 그 속도를 무시해도 될 만큼 빠르기 때문에 신경을 쓰지 않아도 큰 속도 차이 없이 작동한다. 그러나 처리해야 할 스레드의 개수가 매우 많다면 문맥 교환이 매우 빈번하게 발생하여 속도 저하가 발생한다.

이처럼 문맥 교환 작업은 매우 미묘하더라도 멀티 스레딩(multi threading)을 해야 하는 상황에서는 스레드의 개수에 신경을 써야 한다. 병렬 처리로 작업 속도를 증가시킬 수는 있지만, 개수가 너무 많아지면 오히려 역효과를 가져올 수도 있으므로, 성능 테스트를 반드시 거쳐야 한다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원

[그림 3-5] 문맥 교환(context switching)

3. 임계 구역(critical section)

임계 구역은 멀티 스레딩에서 동기화를 하기 위한 수단이다. 하나의 객체나 루틴에 여러 스레드가 동시에 접근할 때 오류가 발생할 우려가 있는 곳을 임계 구역으로 지정한다. 임계 구역은 엔터 크리티컬 섹션(enter critical section; 임계 구역 진입)으로 진입하며, 동시에 1개의 스레드에만 접근할 수 있고 나머지 스레드는 대기한다. 임계 구역의 끝은 리브 크리티컬 섹션(leave critical section; 임계 구역 끝)으로 지정한다.

4. 교착 상태(dead lock)

임계 구역에 진입한 스레드가 리브 스레드 섹션(leave critical section; 임계 구역 끝)을 호출하지 않았다면, 접근을 시도하는 다른 모든 스레드들은 임계 구역에 진입할 수 없다. 이렇게 예기치 않은 실수로 모든 스레드들이 멈춰 버리는 현상이 발생하는데, 이를 교착 상태(dead lock)라고 한다.

5. 병목 현상

임계 구역(critical section)으로 구성된 임계 구역은 동시에 1개의 스레드만 진입할 수 있다. 예를 들어, 5개의 스레드가 진입을 시도하면 그중 1개만 진입하고, 나머지 4개는 대기해야 하는 병목 현상이 발생한다.

6. 입출력 완료 포트 구조(IOCP)

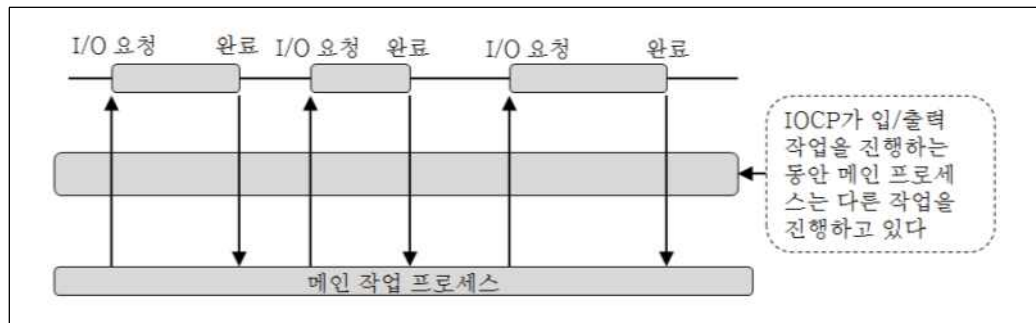
(1) 정의

IOCP는 'input output completion port'의 약자로, 단어의 뜻 그대로 '입출력 완료 포트'를 말하며, 입출력 작업을 처리해 주는 역할을 한다. 응용 프로그램은 사용자나 파일 등으로부터 자료의 입력(input)을 받으며, 그 결과를 파일과 화면 등의 장치로 출력(output)한다. 이러한 입출력 작업들을 좀 더 편리하고 효율적으로 처리할 수 있도록 도와주는 도구가 바로 IOCP이다.

(2) 동작

IOCP는 메인 프로세스와 별도로 동작한다. 이는 현재 입출력 작업이 어떻게 진행되고 있는지 작업 프로세스에서는 알 필요가 없으며, 단지 작업의 결과만을 통보받겠다는 의미이다. 이렇듯 관련된 2개 이상의 작업이 병렬적으로 발생하여 진행되는 과정을 '비동기'라고 한다.

이처럼 IOCP는 작업을 비동기로 처리하며, 이와 동시에 여러 개의 핸들 작업을 할 수 있다. 다시 말해, 하나의 IOCP 객체로 여러 개의 파일에 대한 작업을 수행할 수 있다는 것이다.



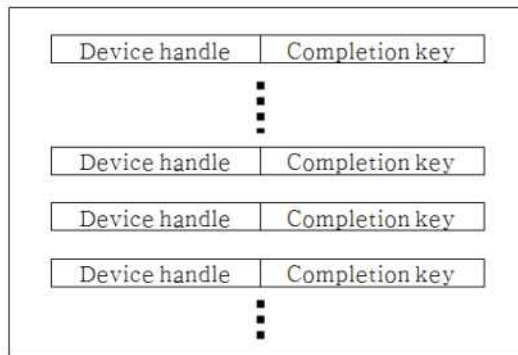
출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원

[그림 3-6] 비동기 작업 프로세스

(가) 디바이스 리스트(device list)

여러 개의 핸들로 작업할 때 결과 통보만을 받아서 어느 핸들의 작업이 완료된 것인지 어떻게 판단할 수 있을까? 이 문제에 대한 해결책으로 IOCP에서는 완료 키(completion key) 값을 지원한다.

처음 IOCP에 핸들을 등록할 때에는 DWORD형으로 완료 키 값을 함께 등록한다. 이렇게 핸들 하나와 완료 키 하나가 쌍을 이루어 IOCP에 작업을 요청하는데, 새로운 작업이 요청될 때마다 IOCP 내부에서는 이 값들을 따로 저장한다. 이 부분의 자료 구조를 디바이스 리스트(device list)라고 한다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원

[그림 3-7] 디바이스 리스트(device list)

(나) IOCP의 완료 큐

IOCP는 이렇게 등록된 핸들 가운데 입출력 이벤트가 발생한 핸들을 처리하고, 메인 프로세스에는 핸들에 대응되는 완료 키와 함께 다른 관련 정보들을 넘겨준다. 결과를 통보할 때 해당 핸들은 넘겨받지 않으므로 DWORD형으로 정의된 완료 키 값을 사용해야 한다. 완료 키로 핸들을 지정해 넣을 수도 있지만, 일반적으로는 핸들과 함께 관련 정보를 구조체로 저장하여 그 포인터를 넘겨주는 방법을 사용한다.

(다) 큐 저장 방식

IOCP는 이벤트나 메시지 등의 방법으로 결과를 통보하지 않는다. 구체적으로 말하자면, 결과에 대한 정보를 IOCP 내부에서 큐로 저장하며, 순차적으로 저장된 큐에서 정보를 받아오는 것이다. 이 큐는 FIFO(first in first out)의 형태이며, 언제 결과를 통보받을지 모르기 때문에 이 큐를 항상 감시해야 한다. 보통 네트워킹과 같이 빈번히 발생하는 작업의 경우에는 이 큐를 감시하는 스레드를 따로 생성하지만, 순간적인 입출력 작업에서는 메인 프로세스에서 처리하는 방법을 사용한다.

수행 내용 / 네트워크 패킷 구조 재구축하기

재료·자료

- 프로토콜 설계 문서, 전송 패킷 구조도, 온라인 게임 서버 구조도, 게임 기획서, 소켓 프로그래밍 참고서, IOCP 관련 참고서

기기(장비·공구)

- 컴퓨터, VOOOOO SOOOOO 2019 이상 버전, MyOOO 데이터베이스 설치 프로그램

안전·유의 사항

- 2인 1조로 팀을 구성하여 클라이언트, 서버의 역할을 수행한다.
- 코드 구현과 실행 시 프로그램이 비정상적으로 종료될 수 있기 때문에 저장과 관리에 항상 유의한다.

수행 순서

① 서버 프로그램을 작성한다.

1. 단순한 서버와 클라이언트 연동을 위한 TCP/UDP 서버 프로그램을 제작하기 위해 서버 프로그램의 순서도 또는 의사 코드를 작성한다.
2. 새로운 TCP/UDP 소켓을 생성하는 코드를 작성한다.

〈표 3-3〉 TCP/UDP 소켓 생성 코드 예

```
01  socketclient = INVALID_SOCKET;
02  socketlisten = ::socket(PF_INET, SOCK_STREAM, IPPROTO_TCP);
03  if(socketlisten == INVALID_SOCKET)
04  {
05      printf("TCP Socket create error !!\n");
06      exit (-1);
07  }
```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

‘::socket’은 소켓 생성 핸들로 파라미터는 ‘PF_INET’는 ‘IPv4’ 타입을 사용한다는 정의이며, ‘SOCK_STREAM’은 연결 지향형 소켓을 만들겠다는 의미이다. ‘IPPROTO_TCP’는 TCP를 지정해 주는 코드이다.

〈표 3-4〉 UDP 소켓 생성 코드 예

```

01 int socketlisten;
02 socketlisten = socket( PF_INET, SOCK_DGRAM, 0);
03 if(socketlisten == -1)
04 {
05     printf("UDP Socket create error !!\n");
06     exit (-1);
07 }

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

‘socket’은 TCP와 동일하게 소켓 생성 핸들로 ‘IPv4’를 사용하며, ‘SOCK_DGRAM’은 UDP 프로토콜 전송 방식을 사용하겠다는 정의이다. 이때, 그 값으로 ‘0’ 또는 ‘IPPROTO_UDP’를 사용하면 UDP 프로토콜 통신 방식을 사용한다고 지정하는 코드이다.

3. 생성된 TCP/UDP 소켓에 지정된 주소로 연결하는 코드를 작성한다.

〈표 3-5〉 TCP 리슨(TCP listen) 소켓 생성 코드 예

```

01 ::memset(&serverAddr, 0, sizeof(serverAddr));
02 serverAddr.sin_family = AF_INET;
03 serverAddr.sin_addr.s_addr = ::htonl(INADDR_ANY);
04 serverAddr.sin_port = ::htons(8600);
05 if(::bind(socketlisten, (struct sockaddr*)&serverAddr,sizeof(serverAddr)) ==
06     SOCKET_ERROR)
07 {
08     printf("bind failed!! : %d\n", ::WSAGetLastError() );
09     return;
10 }
11 if(::listen(socketlisten, SOMAXCONN) == SOCKET_ERROR)
12 {
13     printf("listen failed!! : %d\n", ::WSAGetLastError() );
14     return;
15 }

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

〈표 3-6〉 UDP 리슨(UDP listen) 소켓 생성 코드 예

```

01 struct sockaddr_in server_addr;
02 memset(&server_addr, 0, sizeof(server_addr));
03 server_addr.sin_family = PF_INET;
04 server_addr.sin_port = 10000
05 server_addr.sin_addr.s_addr = htons("192.168.0.100");
06 socketlisten = bind (sock, (struct sockaddr*)&server_addr, sizeof( server_addr) );
07 if ( socketlisten == -1)
08 {
09     printf("listen failed!!");
10     exit (-1);
11 }

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

TCP 프로토콜의 통신 특성상 서버에서 접속 요청을 하면 클라이언트가 통신 확인 응답을 주고받는 '3-way handshake' 방식으로 서버와 클라이언트 간의 접속 요청을 처리해야 한다.

〈표 3-7〉 TCP 접속 요청 처리 코드 예

```

01 fd_set fds;
02 struct timeval tv = { 0,100 };
03 FD_ZERO(&fds);
04 FD_SET(socketListen, &fds);
05 ::select( 0, &fds, 0, 0, &tv);
06 if(FD_ISSET( socketListen, &fds))
07 {
08     struct sockaddr_in fromAddr;
09     int size = sizeof(fromAddr);
10
11     socketClient = ::accept( socketListen, ( struct sockaddr* )&fromAddr, &size );
12     printf("Accept Client : %s\n", ::inet_ntoa(fromAddr.sin_addr));
13 }

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

4. 서버와의 통신을 반복 수행하는 코드를 작성한다.

〈표 3-8〉 TCP에서의 송·수신 작업 코드 예

```

01 char recvBuffer[127];
02 int recvBytes;
03 recvBytes = ::recv(socketClient, recvBuffer, 127, 0);
04 printf("%d bytes received : %s\n", recvBytes, recvBuffer );
05 ::send(socketClient, recvBuffer, recvBytes, 0);

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

5. 연결을 끊고 소켓을 종료하는 코드를 작성한다.
6. 프로그램을 실행하고 필요 시 디버깅 작업을 수행한다.
7. 개발 프로그램 문서와 개발 소스 코드 주석을 정리한다.
8. 개발 프로그램의 향후 수정과 응용 방향을 설정한다.

수행 tip

- 서버 프로그램의 작성은 단계와 절차별로 실습을 진행한다. 충분한 단계와 절차를 이해한 뒤, 자신의 게임 기획에 맞게 서버를 구성한다.
- TCP와 UDP 간의 소켓 동작의 차이점을 충분히 이해한 뒤 실습을 진행한다.

② 멀티 스레딩(multi threading) 기반 서버 프로그램을 작성한다.

1. 스레드 생성 코드를 구현한다.

〈표 3-9〉 스레드 생성 코드 예

```
01 class Thread{  
02     public:  
03         class id;  
04         ~start();  
05     }  
    Thread th = new Thread();  
    th.start();
```

2. 스레드 종료 코드를 구현한다.

3. 스레드 클래스(thread class)를 구현한다.

4. 스레드 매니저(thread manager)를 구현한다.

5. 동기화 객체를 구현한다.

(1) 객체의 동시 접근 코드를 구현한다.

(2) 임계 구역(critical section) 코드를 구현한다.

(3) 이벤트(event) 객체 코드를 구현한다.

6. 테스트 및 디버깅을 한다.

③ 서버 프로그램에 IOCP를 적용한다.

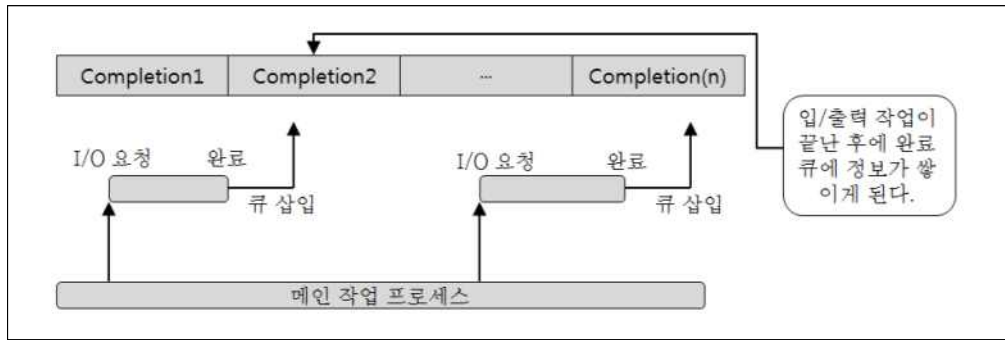
1. 서버 프로그램에 IOCP 코드 추가한다.

(1) 소켓 등록 기능을 구현한다.

(2) 패킷 전송 기능을 구현한다.

(3) 패킷 수신 기능을 구현한다.

2. 완료 큐의 처리 방법을 구현한다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

[그림 3-8] IOCP의 완료 큐

3. 작업 스레드(worker thread) 기능을 구현한다.
4. 데이터 전송 테스트와 디버깅을 수행한다.

수행 tip

- 멀티 스레드를 이해하기 위해 실습 주제별 단편적인 코드로 실습을 진행한다.
- 수행 순서별 각 단계에서 멀티 스레드의 이해와 알고리즘을 충분히 파악할 수 있도록 동일 주제를 반복하여 실습한다.

교수 방법

- 데이터 통신의 이론적인 지식을 충분히 전달하고 이해시킨 뒤 단계적인 실습을 지도한다.
- 네트워크 프로토콜 구현 학습은 기초적인 서버와 클라이언트 간의 데이터 전송과 수신의 이해 그리고 서버와 클라이언트 간의 소켓 연결 및 연결 상태 유지를 중심으로 지도한다.
- 실습은 2인 1조 또는 그 이상의 인원으로 한 팀을 구성하여 서버와 클라이언트의 역할을 각각 담당시킨 뒤 실습을 지도한다.
- 실습 중 서버와 클라이언트의 구현 작업은 팀과 상관없이 학습자 개인별로 서버 프로그램과 클라이언트 프로그램을 모두 구현하되, 서버 프로그램과 클라이언트 프로그램의 결과 확인은 2인 1조로 수행하도록 지도한다.

학습 방법

- 데이터 통신에 관련된 용어를 이해하고, 서버와 클라이언트 간의 데이터 전송과 수신 체계를 이해하는 데 중점을 둔다.
- 게임 서버군의 각 서버 간 전송 데이터와 수신 데이터의 흐름을 중심으로 게임 서버의 역할을 이해한다.
- 데이터 통신의 주된 프로토콜인 TCP를 이해하고, TCP에서 지원하는 다양한 소켓 옵션을 이해한다.

학습 3 평 가

평가 준거

- 평가자는 학습자가 학습 목표를 성공적으로 달성하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가해야 한다.

학습 내용	학습 목표	성취수준		
		상	중	하
TCP 소켓 통신 네트워크 모듈 설계	- TCP/IP 소켓 통신을 활용한 네트워크 모듈을 제작할 수 있다.			
네트워크 패킷 구조 재구축	- 게임 루프와 별개로 돌아가는 네트워크 스레드를 구현할 수 있다.			
	- 패킷 구조를 설계하고 암호화 기법을 활용하여 패킷의 보안을 강화할 수 있다.			
	- 패킷 부하 테스트를 통해 네트워크 모듈의 안정성을 검증하고 최적화할 수 있다.			
	- 패킷 구조와 크기를 서버 트래픽을 고려하여 구현할 수 있다.			

평가 방법

- 포트폴리오

학습 내용	평가 항목	성취수준		
		상	중	하
TCP 소켓 통신 네트워크 모듈 설계	- 서버 프로그램 능력			
	- 클라이언트 프로그램 능력			
	- 프로그램 개발에 대한 문서화 능력			
네트워크 패킷 구조 재구축	- 패킷 구조 설계에 대한 문서화 능력			
	- 패킷 부하 테스트에 대한 문서 작성 능력			

• 서술형 시험

학습 내용	평가 항목	성취수준		
		상	중	하
TCP 소켓 통신 네트워크 모듈 설계	- 서버 프로그램의 순차적인 절차와 기능을 명확하게 숙지하였는지 여부			
	- 클라이언트 프로그램의 순차적인 절차와 기능을 명확하게 숙지하였는지 여부			
	- 개발 프로그램의 문서화가 왜 필요한지 이해하고 있는지 여부			
네트워크 패킷 구조 조재구축	- 패킷 구조 설계에 대한 문서화 능력			
	- 패킷 부하 테스트에 대한 문서 작성 능력			

• 평가자 체크리스트

학습 내용	평가 항목	성취수준		
		상	중	하
TCP 소켓 통신 네트워크 모듈 설계	- TCP 프로토콜에 대해 이해하고 있는지 여부			
	- 서버 프로그램과 클라이언트 프로그램 간의 통신을 위한 순차적 역할을 이해하고 있는지 여부			
	- 개발 프로그램에 대한 문서화 능력			
네트워크 패킷 구조 재구축	- 패킷 구조 설계 능력			
	- 패킷 부하 테스트 과정 설계 능력			

피드백

- 포트폴리오
 - 서버 프로그램에 대한 모범적인 서버 프로그램의 소스 코드를 제시하고, 문법적·논리적 오류를 설명한다.
 - 클라이언트 프로그램에 대한 모범적인 서버 프로그램의 소스 코드를 제시하고, 문법적·논리적 오류를 설명한다.
 - 프로그램 개발 문서 검토 후 문서화의 오류와 누락, 잘못된 내용을 분석하여 설명한다.
- 서술형 시험
 - 개발 프로그램에 대한 문서화의 필요성에 대해 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 다시 설명해 준다.
 - 패킷 구조 설계를 통한 서버의 구조 파악 능력을 평가한 뒤, 평가 결과가 우수한 학습자에 대해서는 패킷 구조를 최적화하는 방법에 대해 설명해 준다.
 - 평가 결과, 소켓 통신 구조에 대한 설계 문서가 잘 정리되어 있는 학습자에게는 서버 구축을 할 수 있도록 지도해 준다.
- 평가자 체크리스트
 - TCP/IP 프로토콜을 정확히 이해할 수 있도록 표와 블록 다이어그램으로 체크한다.
 - 서버/클라이언트 간의 통신 정의와 역할, 순서에 대하여 체크한다.
 - 개발 프로그램 문서가 동작 흐름에 맞게 충분히 설명되었는지 체크한다.

학습 1	게임 서버 시스템 설계하기
학습 2	서버 네트워크 프로그래밍하기
학습 3	C/S 모델에서의 클라이언트 네트워크 프로그래밍하기
학습 4	P2P 모델의 클라이언트 네트워크 프로그래밍하기
학습 5	게임 서버 응용 프로그래밍하기
학습 6	플레이(캐릭터) 프로그래밍하기
학습 7	지형 프로그래밍하기
학습 8	이벤트 프로그래밍하기

4-1. UDP 소켓 통신 네트워크 모듈 기획 및 구축

학습 목표

- UDP 통신을 활용한 네트워크 모듈을 제작할 수 있다.
- 게임 루프와 별개로 돌아가는 네트워크 스레드를 구현할 수 있다.
- 패킷 구조를 설계하고 암호화 기법을 활용하여 패킷의 보안을 강화할 수 있다.
- 패킷 부하 테스트를 통해 네트워크 모듈의 안정성을 검증하고 최적화할 수 있다.
- 개별 피어(peer)로부터 온 패킷을 동기화할 수 있다.

필요 지식 /

① UDP(user datagram protocol)

사용자 데이터그램 프로토콜은 인터넷 프로토콜 스위트(suite)의 주요 프로토콜 가운데 하나이다. 1980년에 데이빗 리드가 설계하였고, 현재 IETF의 RFC 768이 표준으로 정의되어 있으며, TCP와 함께 데이터그램으로 알려진 단문 메시지를 교환하기 위해 사용된다. UDP는 유니버설 데이터그램 프로토콜(universal datagram protocol)이라고도 한다.

UDP의 전송 방식은 너무 단순해서 서비스의 신뢰성이 낮고, 데이터그램 도착 순서가 바뀌거나, 중복되거나, 심지어는 통보 없이 누락시키기도 한다. UDP는 일반적으로 오류 검사와 수정이 필요 없는 애플리케이션(application)에서 수행하는 것으로 가정한다.

UDP를 사용하는 네트워크 애플리케이션에는 도메인 이름 서비스(DNS), IPTV, 음성 인터넷 프로토콜(VoIP), TFTP, IP 터널, 그리고 많은 온라인 게임 등이 있다.

수행 내용 / UDP 소켓 통신 네트워크 모듈 기획 및 구축하기

재료·자료

- 프로토콜 설계 문서, 전송 패킷 구조도, 온라인 게임 서버 구조도, 게임 기획서
- 소켓 프로그래밍 참고서, IOCP 관련 참고서

기기(장비·공구)

- 컴퓨터, VOOOOO SOOOOO 2019 이상 버전, MyOOO 데이터베이스 설치 프로그램

안전·유의 사항

- 2인 1조로 팀을 구성하여 클라이언트와 서버의 역할을 수행한다.
- 코드 구현과 실행 시 프로그램의 비정상 종료가 발생할 수 있으므로 저장과 관리에 항상 유의한다.

수행 순서

① 클라이언트 네트워크 모듈을 작성한다.

1. 단순한 서버와 클라이언트 연동을 위한 TCP 클라이언트 프로그램을 제작하기 위해 기존에 제작된 게임 클라이언트 프로그램의 순서도 또는 의사 코드를 분석하고, 네트워크 설계에 맞게 통신 루틴을 추가한다.

기존에 제작된 게임 클라이언트 프로그램이 없을 경우, 간단히 서버와 연동하여 송·수신된 메시지를 출력할 수 있는 클라이언트 프로그램을 작성한다.

2. 클라이언트 프로그램 내부에 새로운 소켓 생성 기능을 추가로 구현한다.

〈표 4-1〉 클라이언트 TCP 소켓 생성 코드의 예

```
01 socketconnect = ::socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);
02 if(socketconnect == INVALID_SOCKET )
03 {
04     printf("Cannot create socket !!\n");
05     return;
06 }
```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원.

3. 클라이언트 프로그램 내부에 서버 연결 요청 기능을 추가로 구현한다.

〈표 4-2〉 TCP 서버와 연결하기 코드의 예

```
01  ::memset(&serverAddr, 0, sizeof(serverAddr));
02  serverAddr.sin_family = AF_INET;
03  serverAddr.sin_addr.s_addr = ::inet_addr("127.0.0.1");
04  serverAddr.sin_port = ::htons(8600);
05  if(::connect(socketconnect, (struct sockaddr*)&serverAddr, sizeof(serverAddr)) ==
06      SOCKET_ERROR )
07  {
08      printf("Cannot connect to server!!\n");
09      return;
10  }
```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

4. 클라이언트 프로그램 내부에 서버와의 통신을 반복 수행하는 기능을 추가로 구현한다.

설계된 게임 프로토콜 문서를 토대로 게임 내에서 수행해야 할 절차별로 서버에서는 게임 프로토콜을 전송하고, 클라이언트에서는 이 프로토콜을 받아 처리하는 루틴을 작성한다.

〈표 4-3〉 서버와의 데이터 통신 코드의 예

```
01  char sendBuffer[127] = "Test client message...";
02  char recvBuffer[127];
03  int sentBytes, recvBytes;
04  sentBytes = ::send(socketconnect sendBuffer, ::strlen(sendBuffer) + 1, 0);
05  recvBytes = ::recv(socketconnect, recvBuffer, 127, 0 );
```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

5. 클라이언트 프로그램 내부에 서버와의 연결을 끊고 소켓을 종료하는 기능을 추가로 구현한다.

〈표 4-4〉 서버와의 통신 종료 코드의 예

```
01  ::shutdown(socketconnect, SD_BOTH );
02  ::closesocket(socketconnect);
03  ::WSACleanup();
```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

6. 데이터 전송 테스트와 디버깅을 수행한다.

7. 개발 프로그램 문서와 개발 소스 코드의 주석을 정리한다.

8. 개발 프로그램의 향후 수정과 응용 방향을 제시한다.

② 클라이언트 프로그램을 작성한다.

1. 네트워크 모듈을 작성한다.

기존에 제작된 게임 클라이언트 프로그램이 있을 경우, 네트워크 데이터 전송과 수신 모듈을 작성하여 추가한다.

기존에 제작된 게임 클라이언트 프로그램이 없을 경우, 텍스트 기반의 이동과 공격, 아이템 획득이 가능한 간단한 클라이언트 프로그램을 제작한 다음 클라이언트 네트워크 모듈을 작성하여 추가한다.

2. 클라이언트 패킷(client packet) 모듈을 구현한다.

(1) ‘1-1. 게임 서버 시스템 파악’ 학습내용에서 설계한 패킷 설계 문서를 토대로 패킷 클래스(packet class)를 구현한다.

(2) 패킷 클래스에 패킷 데이터의 조립과 패킷 데이터의 분해 모듈을 구현한다.

(3) 패킷 데이터의 전송과 수신 모듈을 작성한다.

3. 클라이언트 패킷 모듈에 패킷의 암호화 모듈을 추가한다.

(1) XOR 기반 암호화 모듈 또는 DES 기반 암호화 모듈 알고리즘을 패킷 데이터 조립 모듈에 적용한다.

(2) 패킷 데이터 전송 모듈을 수정하여 암호화된 패킷 데이터의 전송 모듈을 작성한다.

교수 방법

- 데이터 통신의 이론적인 지식을 충분히 전달하고 이해시킨 다음 단계적인 실습을 지도한다.
- 네트워크 프로토콜 구현 학습은 기초적인 서버와 클라이언트 간의 데이터 전송과 수신의 이해, 그리고 서버와 클라이언트 간의 소켓 연결과 연결 상태 유지를 중심으로 지도한다.
- 실습은 2인 1조 또는 그 이상의 인원을 한 팀으로 구성하여 서버와 클라이언트의 역할을 각각 담당시킨 후 실습을 지도한다.
- 실습 중 서버와 클라이언트의 구현 작업은 팀과 상관없이 학습자 개인별로 서버 프로그램과 클라이언트 프로그램을 모두 구현하되, 서버 프로그램과 클라이언트 프로그램의 결과 확인은 2인 1조로 수행하도록 지도한다.

학습 방법

- 데이터 통신에 관련된 용어를 이해하고, 서버와 클라이언트 간의 데이터 전송과 수신 체계를 이해하는 데 중점을 둔다.
- 게임 서버군의 각 서버 간 전송 데이터와 수신 데이터의 흐름을 중심으로 게임 서버의 역할을 이해한다.
- 데이터 통신의 주된 프로토콜인 TCP를 이해하고, TCP에서 지원하는 다양한 소켓 옵션을 이해한다.

평가 준거

- 평가자는 학습자가 학습 목표를 성공적으로 달성하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가해야 한다.

학습 내용	학습 목표	성취수준		
		상	중	하
UDP 소켓 통신 네트워크 모듈 기획 및 구축	- UDP 통신을 활용한 네트워크 모듈을 제작할 수 있다.			
	- 게임 루프와 별개로 돌아가는 네트워크 스레드를 구현할 수 있다.			
	- 패킷 구조를 설계하고 암호화 기법을 활용하여 패킷의 보안을 강화할 수 있다.			
	- 패킷 부하 테스트를 통해 네트워크 모듈의 안정성을 검증하고 최적화할 수 있다.			
	- 개별 피어(peer)로부터 온 패킷을 동기화할 수 있다.			

평가 방법

- 포트폴리오

학습 내용	평가 항목	성취수준		
		상	중	하
UDP 소켓 통신 네트워크 모듈 기획 및 구축	- 네트워크 모듈 제작 능력			
	- 패킷 동기화 모듈 구축 능력			
	- 패킷 부하 테스트를 통한 검증 능력			

- 서술형 시험

학습 내용	평가 항목	성취수준		
		상	중	하
UDP 소켓 통신 네트워크 모듈 기획 및 구축	- 서버 프로그램의 순차적인 절차와 기능을 명확하게 숙지하였는지 여부			
	- 클라이언트 프로그램의 순차적인 절차와 기능을 명확하게 숙지하였는지 여부			
	- 네트워크 스레드의 구조 파악 능력			

• 평가자 체크리스트

학습 내용	평가 항목	성취수준		
		상	중	하
UDP 소켓 통신 네트워크 모듈 기획 및 구축	- UDP 프로토콜에 대해 이해하고 있는지 여부			
	- 서버 프로그램과 클라이언트 프로그램 간의 통신을 위한 순차적 역할을 이해하고 있는지 여부			
	- 네트워크 모듈 제작 능력			

피드백

1. 포트폴리오

- 서버 프로그램에 대한 모범적인 서버 프로그램의 소스 코드를 제시하고, 문법적 · 논리적 오류를 설명해 준다.
- 클라이언트 프로그램에 대한 모범적인 서버 프로그램의 소스 코드를 제시하고, 문법적 · 논리적 오류를 설명해 준다.
- 프로그램 개발 문서 검토 후 문서화의 오류나 누락, 잘못된 내용을 분석하여 설명해 준다.
- 패킷 구조에 대해서 충분히 익힌 학습자들에게는 보완과 안정성에 대해 실습할 수 있도록 지도해 준다.

2. 서술형 시험

- 개발 프로그램의 문서화의 필요성과 역할에 대해 평가한 뒤, 평가 결과가 미흡한 학습자에게는 다시 설명해 준다.
- 서버와 클라이언트 간의 통신에 대한 이해도를 평가한 뒤, 이해도가 높은 학습자에 대해서는 네트워크 구조에 따른 부하를 테스트하는 방법을 설명해 준다.

3. 평가자 체크리스트

- UDP 프로토콜을 정확히 이해할 수 있도록 표와 블록 다이어그램으로 체크한다.
- 서버/클라이언트 간의 통신 정의와 역할, 순서에 대해 체크한다.
- 개발 프로그램 문서가 동작 흐름에 맞게 충분히 설명되었는지 체크한다.

학습 1	게임 서버 시스템 설계하기
학습 2	서버 네트워크 프로그래밍하기
학습 3	C/S 모델에서의 클라이언트 네트워크 프로그래밍하기
학습 4	P2P 모델의 클라이언트 네트워크 프로그래밍하기
학습 5	게임 서버 응용 프로그래밍하기
학습 6	플레이(캐릭터) 프로그래밍하기
학습 7	지형 프로그래밍하기
학습 8	이벤트 프로그래밍하기

5-1. 게임 서버 구조 설계

학습 목표

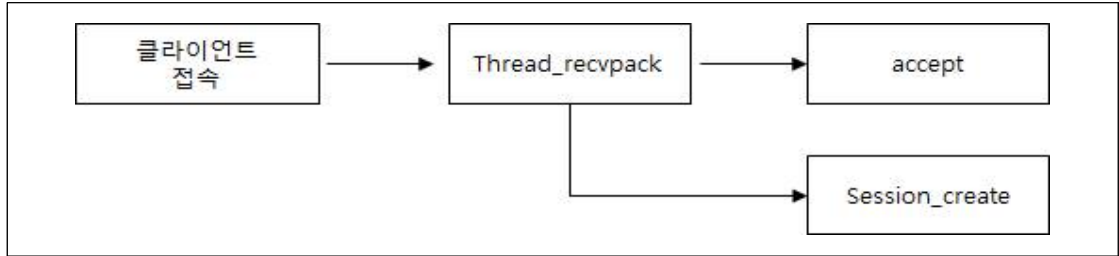
- 게임 기획을 바탕으로 서버와 DB, 서버와 클라이언트 간의 데이터 전송 구조를 설계하고 알고리즘을 작성할 수 있다.
- 작성한 알고리즘에 따라 서버 플랫폼에 적합한 프로그래밍 언어로 작성할 수 있다.

필요 지식 /

‘온라인 RPG 게임’ 장르는 다양한 게임 장르의 구현에 필요한 대부분의 기술들을 요구하므로, 이 학습 모듈은 실제 교육 현장에서 산출될 다양한 장르의 게임 기획서에 대응할 수 있도록 기술의 포괄적 차원에서 ‘온라인 RPG 게임’이라는 가상의 기획 장르를 설정하여 학습을 수행하고 있다.

① 클라이언트의 게임 서버 접속 절차

클라이언트가 게임 서버에 접속하면 스레드 리시브팩(thread_recvpack)에서는 스레드에 의해 새로운 접속이 생겼음을 알게 된다. 소켓 라이브러리(socket library)는 새로운 접속을 받아들이고, 서버에서는 새로운 연결을 위해 연결 정보를 생성한다. 연결 정보가 생성되는 시점부터 서버/클라이언트 사이의 패킷 통신이 가능하므로, 서버에서는 연결 정보를 어느 클라이언트가 패킷을 전송하는지를 통해 알 수 있다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원
[그림 5-1] 클라이언트의 게임 서버 접속 절차

② 패킷 암호화 설계와 클라이언트 설계

1. 보안의 필요성

온라인 게임 초창기 시절에는 보안 개념이 희박하였고, 보안에 대해 검증할 수 있는 성능의 하드웨어가 개발되지 않았었다. 21세기 들어 클라이언트 메모리 해킹, 계정 도용, 패킷 스나이핑(packet sniping), 하이재킹(hijacking) 등 인터넷과 클라이언트에서도 보안 사고가 일어나고 있어 데이터 통신 과정뿐 아니라 클라이언트의 내용을 암호화하여 소프트웨어 내용을 들여다 볼 수 없도록 다양한 기능을 제공하여 보안 사고를 줄일 수 있는 방법의 필요성이 대두되었다.

2. 보안 방법

초기에는 서버와 클라이언트 통신에서 암호화를 설계할 때 암호화 논리 연산을 클라이언트에서도 독립적으로 행하였으나, 최근에는 이마저도 해커(hacker)가 클라이언트를 분석하기 때문에 게임 경험치나 아이템의 논리 연산도 서버가 행하기도 한다. 게임의 장르와 특성에 따라 적절하게 서버/클라이언트에 배치하고 데이터 비교를 통하는 등 보안 사고를 줄일 수 있는 방법을 모색해야 한다.

③ 인증과 캐릭터 초기화 과정

1. 클라이언트 인증

클라이언트의 접속 절차가 끝나면 인증 패킷을 서버로 전송한다. 인증이 안 된 상태의 연결 정보를 가지고 있는 클라이언트로부터 패킷이 수신되기 때문에 스레드 인증(thr_beforAuth)에서 패킷을 파싱(parsing)하여 작업을 수행한다. 인증 패킷을 수신하면 데이터베이스 유저 인증(db_user_auth)에 의해 데이터베이스로 질의한다.

2. 캐릭터 데이터 초기화

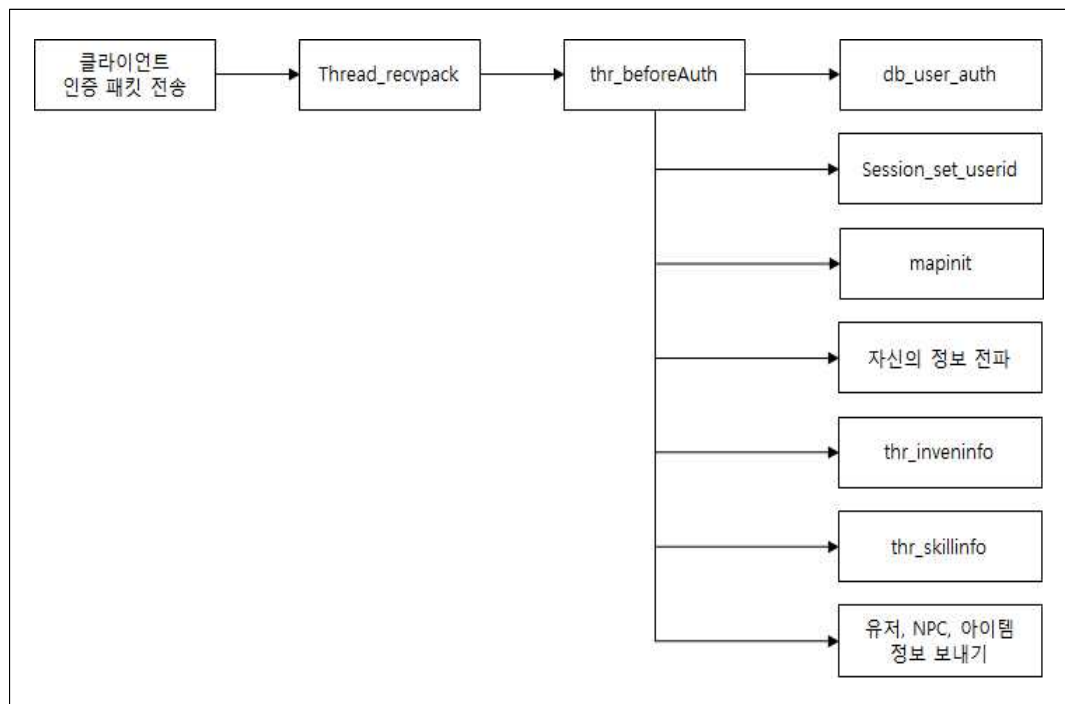
인증이 성공하면 유저 아이디로 클라이언트의 캐릭터 정보를 불러와 연결 정보에 있는 캐릭터 데이터에 초기화하는 작업을 수행하는데, 이때에는 캐릭터의 능력치도 함께 계산한다. 캐릭터의 데이터를 읽고 난 뒤에는 그 데이터에 있는 캐릭터의 위치 좌표를 통해 서버의

맵에 위치시키는 작업을 수행한다. 맵에 위치시키는 것은 맵 섹터(map sector)의 객체 리스트에 캐릭터 객체를 추가하는 작업을 의미한다.

3. 게임 접속

캐릭터를 게임 서버 환경에 추가하는 작업이 끝나면, 클라이언트의 환경에 새로운 객체가 추가되었음을 알린다. 먼저, 새로 추가되는 캐릭터의 정보를 자기 자신과 동기화 영역 내의 캐릭터에게 전송한다. 새로운 캐릭터는 자신의 데이터를 받아 자신의 캐릭터 정보를 적용하고, 다른 클라이언트는 새로운 객체가 추가되는 것을 알게 되어 화면에 표시한다.

새로운 캐릭터는 캐릭터의 기본 정보 외에도 인벤토리(inventory) 정보와 스킬 정보를 따로 전송받고, 마지막으로 캐릭터의 동기화 영역에 있는 다른 캐릭터 정보와 NPC, 아이템 정보를 수신하면 기본적인 캐릭터의 초기화 작업이 끝난다. 초기화 작업이 끝나면 클라이언트는 자신의 캐릭터를 조작하여 게임을 진행할 수 있다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

[그림 5-2] 인증과 캐릭터 초기화 과정

④ 패킷 프로토콜의 정의

패킷 프로토콜을 정의할 때에는 최소한의 정보로 원하는 의미를 정확하게 전송할 수 있도록 작성해야 한다. 또한, 데이터량을 최소화하기 위해 유저 아이디, 비밀번호, 채팅 메시지 같이 길이가 유동적인 스트링 데이터(string data)는 데이터의 길이 정보를 추가하여 패킷의 해당 길이만큼 데이터를 삽입하는 방법을 사용한다.

⑤ 구조체 패딩

1. 데이터 구조

32비트 환경에서는 메모리에 접근할 때 4바이트(32비트) 단위로 접근한다. 메모리에서 데이터를 읽어올 때 주소를 32비트의 배수 단위로 지정하고, 데이터의 크기도 32비트의 배수 단위로 가져온다. 예를 들어, '0×1003~0×1004'번지에 있는 2바이트의 데이터를 가져오기 위해서 '0×1000'번지에서 4바이트를 가져와 그중 3번(네 번째)에 해당하는 1바이트를 취하고, '0×1004'번지에서 4바이트를 가져와 그중 0번(첫 번째) 1바이트를 취하여 둘을 합치는 과정을 거치게 된다. 이렇게 4바이트 단위의 메모리 입출력에 따라 위의 예와 같은 경우는 불필요한 메모리 액세스가 발생하여 효율성이 떨어지게 된다.

2. 데이터 패딩(data padding)

최적화된 메모리의 사용을 위하여 데이터들을 단위 크기로 정렬(align)하고, 정렬을 위하여 사용하지 않는 임의의 공간을 삽입하게 되는데, 이를 '패딩(padding)'이라고 한다.

32비트 CPU를 사용하는 서버에서는 컴파일러가 4바이트 구조체 패딩을 한다. 컴파일러(compiler)에 의해 패딩이 많이 일어나면 메모리가 낭비되므로, 패딩이 최소한으로 일어나도록 구조체 멤버 변수의 순서를 4바이트별로 정의해야 한다.

⑥ 패킷 프로토콜을 디자인할 때 고려할 사항

패킷 프로토콜을 디자인할 때에는 최소한의 데이터로 그 의미를 정확하게 전달할 수 있는 구조로 디자인하는 것이 중요하다. 불필요하게 전송될 수 있는 데이터는 모두 제거하고, 반드시 필요한 데이터만 전송되도록 디자인해야 한다.

객체 추가나 객체 업데이트와 같이 패킷의 구조가 비슷하거나 묶을 수 있는 패킷은 객체 타입(type) 등을 사용한다. 이러한 패킷을 모두 각각 정의해 준다면 패킷의 개수가 많아져 관리하기 불편할 수 있기 때문이다.

⑦ 기본 데이터 로드(load)

게임을 진행할 때 필요한 기초 데이터는 모두 서버 구동과 함께 서버의 메모리에 적재한다. 이러한 기본 데이터는 주로 배열로 저장하며, 배열 인덱스(index)와 해당 인덱스에 저장된 객체의 아이디 번호를 같게 하여 검색 없이 필요한 데이터를 바로 참조할 수 있는 구조를 사용해야 한다.

수행 내용 / 게임 서버 구조 설계하기

재료·자료

- 온라인 게임 서버 구조도, 게임 기획서, 게임 서버 구축 관련 참고 도서

기기(장비·공구)

- 컴퓨터, 윈도우 10 운영 체제 설치 CD, My○○○ 설치 프로그램, Vis○○○ Stu○○○
○ 최신 버전

안전·유의 사항

- PC에 서버 기능을 설치하기 위해 기존에 설치된 운영 체제에 최소한의 서버 기능을 추가한다.
- 최소 3~5인의 인원을 한 팀으로 구성하여 기획, 클라이언트, 서버의 역할을 수행한다.
- 버그 발생 시 프로그램이 비정상적으로 종료되는 경우가 있기 때문에 소스 코드의 저장과 관리에 유의한다.

수행 순서

① 게임 서버 설계와 패킷 전송 흐름을 설계한다.

1. 게임 서버 구조를 설계한다.
2. 패킷 전송 흐름을 설계한다.
3. 패킷 암호화 방식을 설계한다.

② 패킷을 설계한다.

1. 인증 패킷을 설계한다.
2. 채팅 패킷을 설계한다.
3. 인벤토리(inventory) 정보 패킷을 설계한다.
4. 파티(party) 관련 패킷을 설계한다.
 - (1) 파티 요청 패킷을 설계한다.
 - (2) 파티 요청 허락 패킷을 설계한다.
 - (3) 파티 탈퇴 패킷을 설계한다.

- (4) 파티 깨짐 패킷을 설계한다.
 - (5) 파티원(party member) 추방 패킷을 설계한다.
 - (6) 파티 채팅 패킷을 설계한다.
 - (7) 파티 정보 패킷을 설계한다.
5. 캐릭터 관련 패킷을 설계한다.
- (1) 이동 명령 패킷을 설계한다.
 - (2) 공격 명령 패킷을 설계한다.
 - (3) 레벨 업 패킷을 설계한다.
 - (4) 스킬 배우기 패킷과 스킬 레벨 업 패킷을 설계한다.
 - (5) 스킬 정보 패킷을 설계한다.
6. 객체 관련 패킷을 설계한다.
- (1) 객체 추가 패킷을 설계한다.
 - (2) 객체 업데이트 패킷을 설계한다.
 - (3) 객체 장비 업데이트 패킷을 설계한다.
 - (4) 객체 삭제 패킷을 설계한다.
7. 아이템 관련 패킷을 설계한다.
- (1) 아이템 구입·판매 패킷을 설계한다.
 - (2) 아이템 줍기·버리기 패킷을 설계한다.
 - (3) 아이템 위치 변경과 장착·해제 패킷을 설계한다.
8. 아이템 거래 관련 패킷을 설계한다.
- (1) 거래 요청과 요청 확인 패킷을 설계한다.
 - (2) 아이템 리스트 패킷을 설계한다.
 - (3) 아이템 거래 확인·취소 패킷을 설계한다.
 - (4) 아이템 거래 성공 패킷을 설계한다.
9. 창고 정보 관련 패킷을 설계한다.
- (1) 창고 정보 패킷을 설계한다.
 - (2) 창고 아이템 위치 변경 패킷을 설계한다.
 - (3) 창고 아이템 저장 꺼내기 패킷을 설계한다.
10. 게임 맵 관련 패킷을 설계한다.

- (1) 맵 이동 요청 패킷을 설계한다.
- (2) 맵 이동 서버 접속 인증 패킷을 설계한다.

③ 패킷 설계를 정리하여 문서화한다.

④ 게임 서버 기초 모듈을 제작한다.

- 1. 서버/클라이언트 전송 패킷 프로토콜을 구현한다.
- 2. 게임 기본 데이터 로딩 모듈을 구현한다.

⑤ 서버 모듈을 구현한다.

- 1. 유저의 서버 접속 모듈을 구현한다.
- 2. 인증 모듈을 구현한다.
- 3. 캐릭터 초기화 모듈을 구현한다.
- 4. 캐릭터와 NPC의 이동 처리 모듈을 구현한다.
 - (1) 캐릭터 이동 처리 모듈을 구현한다.
 - (2) NPC 이동 처리 모듈을 구현한다.
- 5. 전투 시스템 모듈을 구현한다.
 - (1) 물리 공격 시스템을 구현한다.
 - (2) NPC의 캐릭터 공격 시스템을 구현한다.
- 6. 게임 아이템 시스템 모듈을 구현한다.
 - (1) 캐릭터 아이템과 필드 아이템 관리 시스템을 구현한다.
 - (2) 상점과 아이템 거래 시스템을 구현한다.
- 7. 스킬과 마법 시스템 모듈을 구현한다.
- 8. 파티 시스템을 구현한다.
 - (1) 파티 플레이 시스템을 구현한다.
 - (2) 파티 관리 시스템을 구현한다.
- 9. 캐릭터의 필드 서버 이동 시스템을 구현한다.
- 10. 데이터 저장 시스템을 구현한다.

⑥ 클라이언트/서버 간의 통신 테스트와 디버깅 테스트를 진행한다.

1. 각 모듈별 동작을 테스트한다.
2. 프로그램 오류에 대한 디버깅을 수행한다.

⑦ 개발 문서와 코드를 정리하고, 방향 제시 문서를 작성한다.

1. 개발 프로그램 문서와 개발 소스 코드의 주석을 정리한다.
2. 개발 프로그램의 향후 수정과 응용 방향을 제시한다.

수행 tip

- 기획서의 장르와 형태에 따라 구현 내용과 설계가 달라질 수 있다. 게임 기획서에 따른 서버와 클라이언트의 연동 형태를 파악하여, 게임 서버 구축에 필요한 핵심 원리 위주로 실습을 하고, 팀 단위로 실습을 진행한다.

5-2. 게임 서버 테스트 환경 구축

학습 목표

- 게임 서버 리포팅 및 패치 프로그램을 작성할 수 있다.
- 게임 서버 테스트 환경을 구축하고 테스트들(CBT, OBT등)을 실시할 수 있다.

필요 지식 /

① CBT / OBT

CBT(closed beta test)와 OBT(open beta test)는 게임을 상용화하기 이전에 품질을 검증할 수 있는 테스트로, 결함이나 사소한 문제들을 유저의 참여로 수정하거나 사항을 평가받는 단계이다.

1. CBT(closed beta test)

제한된 유저가 참여하는 경우를 CBT라고 한다.

2. OBT(open beta test)

불특정 다수를 대상으로 테스트를 진행하는 경우를 OBT라고 한다.

3. FGT(focus group test)

CBT/OBT 이전에 특정 사용자를 대상으로 한 FGT(Focus Group Test)를 진행하는 경우도 있다.

수행 내용 / 게임 서버 테스트 환경 구축하기

재료·자료

- 온라인 게임 서버 구조도, 게임 기획서, 게임 서버 구축 관련 참고 도서

기기(장비·공구)

- 컴퓨터, 윈도우 10 운영 체제 설치 CD, My○○○ 설치 프로그램, V○○○○○○ SOO ○○○ 2019 버전

안전·유의 사항

- PC에 서버 기능을 설치하기 위해 기존에 설치된 운영체제에 최소한의 서버 기능을 추가한다.
- 버그 발생 시 프로그램이 비정상적으로 종료되는 경우가 있기 때문에 소스 코드의 저장과 관리에 유의한다.

수행 순서

① CBT/OBT에 앞서 내부에서 진행할 각 기능별, 모듈별 테스트할 항목을 작성한다.

② 각 기능, 항목에 맞는 테스트를 진행한다.

〈표 5-1〉 각 모듈별 테스트 항목

메인 기능	번호	서브 기능	내용	결과	설명
퀘스트	1	몬스터의 습격1	평온지역 자비스 NPC에게 이야기를 듣고 A 몬스터 10마리를 잡는다.	PASS	
	2	몬스터의 습격 2	평온지역 자비스 NPC에게 이야기를 듣고 B 몬스터 10마리를 잡는다.	PASS	
	3	초보 기사1	몬스터의 습격 2를 완료 후 평온지역 파멜라 NPC에게 이야기를 하면 초보 기사 자격을 받을 수 있는 NPC를 소개시켜 준다.	Fail	상위 퀘스트 완료 후에 퀘스트 습득이 불가
	4	초보 기사1 자격 시험	파멜라에게 소개받은 AAA NPC에게 자격 시험을 듣고 해당 퀘스트를 받는다.	Fail	상위 퀘스트의 실패로 테스트 불가
		⋮	⋮	⋮	
상점 NPC	1	노미스 NPC	방어구 아이템을 판매	PASS Check	각 항목의 아이템 금액은 확인 요망
		⋮	⋮	⋮	

수행 tip

- 실제 테스트용 체크리스트를 작성할 때에는 테스터 및 기획자, 개발자가 정확히 인지할 수 있도록 내용을 추가한다.

- ③ 진행된 테스트가 기획과 일치하는지 체크하고, 발생한 문제점을 수정한다.
- ④ 사용자를 일정하게 모집하여 CBT/OBT를 진행하여 사용자의 버그나 불편 사항을 제공받아 그 테스트 내용을 분석하고 생각하지 못했던 기획의 오류를 바로 잡는다.
- ⑤ CBT 중 기획 내용과 상이하게 동작하거나, 성능상 문제되는 부분은 프로그램한 코드를 최적화하거나 튜닝하는 작업을 진행한다.
- ⑥ CBT에서 문제를 충분히 수정한 뒤, OBT를 하여 서비스 테스트를 진행한다. CBT에서 잡지 못한 버그나 불편 사항들을 사용자에게 제공받아 프로그램을 수정한다.
- ⑦ 개발 문서와 코드를 정리하고, 방향 제시 문서를 작성한다.
 1. 개발 프로그램 문서와 개발 소스 코드의 주석을 정리한다.
 2. 개발 프로그램의 향후 수정과 응용 방향을 제시한다.

수행 tip

- 테스트 항목과 성능 이슈의 복잡도에 따라 프로그램 완성도의 차이가 많이 발생하기 때문에, 시간을 할애하여 상세 항목을 적용한다.

교수 방법

- 데이터 통신의 이론적인 지식을 충분히 전달하고 이해시킨 다음 단계적인 실습을 지도한다.
- 이 학습 모듈은 팀 단위 프로젝트 중심으로 실습을 진행하도록 현장 지도한다.
- 실습의 내용과 순서는 팀 단위 기획서의 내용에 따라 달라질 수 있다. 서버 구축의 합집합 격인 비교적 가장 규모가 큰 MMORPG 서버 구축을 중심으로 교육을 진행하고, 팀의 기획서에 맞게 가감하여 구축하도록 지도한다.
- 이 학습 모듈에서는 네트워크 관련 지식과 서버 구축에 관련된 총괄적인 지식을 요구한다. 따라서 충분한 기초 교육이 이루어진 뒤에 실습을 진행하도록 한다.
- 학습 단위 주제별 실습 예제 코드를 유인물과 소스 코드 공유, 배포 등을 통해 진행할 수 있도록 사전에 준비한다.

학습 방법

- 이 실습 모듈은 네트워크 관련 학습과 서버 구축에 관련된 기초 지식을 요구하고 있다. 따라서 충분한 사전 학습을 수행하고, 이 학습 모듈의 실습을 진행한다.
- 구축한 게임 서버의 구조를 다각면의 기능 테스트를 진행한 뒤 안정화된 서버 구조를 구축할 수 있도록 학습한다.
- 학습의 효율성을 향상시키기 위해서는 게임 서버 구축을 위한 요소별 핵심 원리와 알고리즘 위주로 학습한다.

평가 준거

- 평가자는 학습자가 학습 목표를 성공적으로 달성하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가해야 한다.

학습 내용	학습 목표	성취수준		
		상	중	하
게임 서버 구조 설계	- 게임 기획을 바탕으로 서버와 DB, 서버와 클라이언트 간의 데이터 전송 구조를 설계하고 알고리즘을 작성할 수 있다.			
	- 작성한 알고리즘에 따라 서버 플랫폼에 적합한 프로그래밍 언어로 작성할 수 있다.			
게임 서버 테스트 환경 구축	- 게임 서버 리포팅 및 패치 프로그램을 작성할 수 있다.			
	- 게임 서버 테스트 환경을 구축하고 테스트들(CBT, OBT 등)을 실시할 수 있다.			

평가 방법

- 포트폴리오

학습 내용	평가 항목	성취수준		
		상	중	하
게임 서버 구조 설계	- 게임 기획서에 적합한 게임 서버 설계 능력			
	- 팀 단위 기획서에 적합한 패킷 설계 능력			
게임 서버 테스트 환경 구축	- 게임 서버 구조화 능력			
	- 개발된 프로그램의 문서화 능력			

- 서술형 시험

학습 내용	평가 항목	성취수준		
		상	중	하
게임 서버 구조 설계	- 게임 서버에 대해 이해하고 있는지 여부			
	- 게임 프로토콜 설계와 패킷 설계 능력			
게임 서버 테스트 환경 구축	- 게임 기획서에 기반한 각 항목별 알고리즘 구현 능력			
	- 구현 항목별 기초 이론과 구현 원리에 대해 이해하고 있는지 여부			

• 평가자 체크리스트

학습 내용	평가 항목	성취수준		
		상	중	하
게임 서버 구조 설계	- 게임 서버 구축의 원리와 방법에 대해 이해하고 있는지 여부			
	- 자신의 기획서에 적합한 게임 서버를 설계하는 능력			
	- 게임 서버 구축에 필요한 프로토콜을 설계하는 능력			
	- 합리적으로 패킷을 설계하고 구현하는 능력			
게임 서버 테스트 환경 구축	- 서버와 연동하여 게임 캐릭터를 제어하는 능력			
	- 게임 서버에 사용자의 정보를 저장하고, 클라이언트로 전송받는 능력			
	- 게임 서버와 연동한 게임 데이터를 제어하는 능력			
	- 게임 기획서에 적합한 게임 서버 알고리즘을 구현하는 능력			

피드백

1. 포트폴리오

- 게임 기획서를 바탕으로 한 게임 서버 설계의 흐름을 충실히 이행하고 있는지 평가한 뒤, 평가 결과가 부족한 학습자에 대해서는 설명을 통해 지도한다.
- 게임 서버 설계와 개발된 프로그램 사이의 오류를 체크하고, 부족한 부분은 설명을 통해 지도한다.

2. 서술형 시험

- 게임 기획서를 기반으로 한 각 항목별 구현 알고리즘에 대해 평가한 뒤, 평가 결과가 미흡한 학습자에게는 다시 설명해 준다.

3. 평가자 체크리스트

- 게임 기획을 바탕으로 만든 서버 프로그램 소스를 체크리스트로 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 주요 사항을 표시하여 설명해 준다.

학습 1	게임 서버 시스템 설계하기
학습 2	서버 네트워크 프로그래밍하기
학습 3	C/S 모델에서의 클라이언트 네트워크 프로그래밍하기
학습 4	P2P 모델의 클라이언트 네트워크 프로그래밍하기
학습 5	게임 서버 응용 프로그래밍하기
학습 6	플레이(캐릭터) 프로그래밍하기
학습 7	지형 프로그래밍하기
학습 8	이벤트 프로그래밍하기

6-1. 입출력 데이터에 따른 캐릭터 반응 설계 및 구현

학습 목표

- 사용자가 조작 가능한 입출력 및 캐릭터 반응을 설계할 수 있다.
- 사용자 입출력에 반응한 캐릭터의 동작과 반응을 구현할 수 있다.

필요 지식 /

① 정점 애니메이션 기술

정점 애니메이션 방식은 일종의 모르핑(morphing)으로, 퀘이크(Quake) 시리즈가 대표적인 예이다. 이 방식은 정점의 애니메이션 된 위치를 매 프레임(frame)마다 저장하고 있다가 이것을 선형 보간에 의해 출력하는 것으로, 원리와 구성 자체가 간단하기 때문에 구현하기가 매우 쉬우며, 자동으로 스킨닝(skinning)된다는 장점이 있다. 그러나 데이터의 양이 너무 많다는 문제점이 있다.

〈표 6-1〉 정점 애니메이션 연산식

$$\text{정점 애니메이션 연산식}$$

$$V_{world} = (1 - a) \times V_1 + a \times V_2 \quad (\text{단}, 0 \leq a \leq 1)$$

② 애니메이션 계층 구조

인간과 비슷한 형태의 관절을 가진 오브젝트(articulated body)는 계층 구조로 구성되어 있다. 인간의 경우 허리를 뿌리 노드(node)로 하여 상하체가 붙어 있고, 상체에 목, 어깨, 가슴 등이 붙어 있는 구조이다. 캐릭터를 여러 개의 메시(mesh)로 나눈 다음, 각 메시지를 부모-자식 관계

로 상속시키고, 메시에 이동, 축소, 회전을 포함한 매트릭스(matrix)를 매 프레임 또는 변화되는 프레임마다 저장하여 사용하는 것이다.

디테일 맵(detail map)은 정점에 텍스처(texture)를 입히는 것과 같기 때문에, 텍스처의 UV 좌표가 설정될 수 있도록 <표 6-2>와 같이 정점 구조체와 FVF를 구성한다.

<표 6-2> 정점 구조체와 FVF

```
D3DXMATRIXA16* ZNode::Animate( float fFrame, D3DXMATRIXA16* pParentTM )
{
    m_matTM = m_matLocal * m_matAni * *pParentTM;
    return &m_matTM;
}

int ZNodeMgr::Animate( float fFrame )
{
    int i;
    int id;
    D3DXMATRIXA16* pTM;

    for( i = 0 ; i < m_nodes.size( ) ; i++ )
    {
        id = m_nodes[i]->GetParentID( );
        pTM = m_nodes[id]->GetMatrixTM( );
        ((ZNode *)m_nodes[i])->Animate( fFrame, pTM );
    }

    return 1;
}

VOID Animate( )
{
    ((ZNodeMgr *)g_pNodeMgr)->Animate( 0 );    // 노드 애니메이션
}
```

③ 캐릭터의 이동

캐릭터는 프레임을 바탕으로 이동 카운트(count)를 프레임마다 카운트하여 이동 속도에 맞게 카운트될 때 이동한다. 캐릭터가 게임 맵에서 타일(tile)을 이동할 때마다 현재 위치의 섹터를 검사하여 섹터를 이동했다면 이전 섹터에 있는 자신의 객체를 새로운 섹터로 옮기는 작업을 수행해야 하며, 새로운 섹터의 이동에 따른 동기화 작업을 수행해야 한다. 동기화 작업은 새롭게 캐릭터의 동기화 영역이 되는 섹터의 유저에게 캐릭터 정보를 전송하고, 기존의 객체 정보를 수신하는 방식으로 이루어진다.

④ NPC 이동

기본적인 이동 방식은 캐릭터의 이동 방식과 동일하다. NPC가 순화할 경우에는 패턴을 자신의 생성 지역을 멀리 벗어나지 못하도록 설정해야 한다. 랜덤(random)한 시간 간격과 랜덤한 거리, 랜덤한 방향으로 이동하도록 패턴을 정의하면 좀 더 자연스럽게 이동한다.

수행 내용 / 입출력 데이터에 따른 캐릭터 반응 설계 및 구현하기

재료·자료

- 시스템 기획서, 본 애니메이션(bone animation)이 적용된 캐릭터 모델링 데이터, 모델링에 입힐 텍스처 데이터

기기(장비 · 공구)

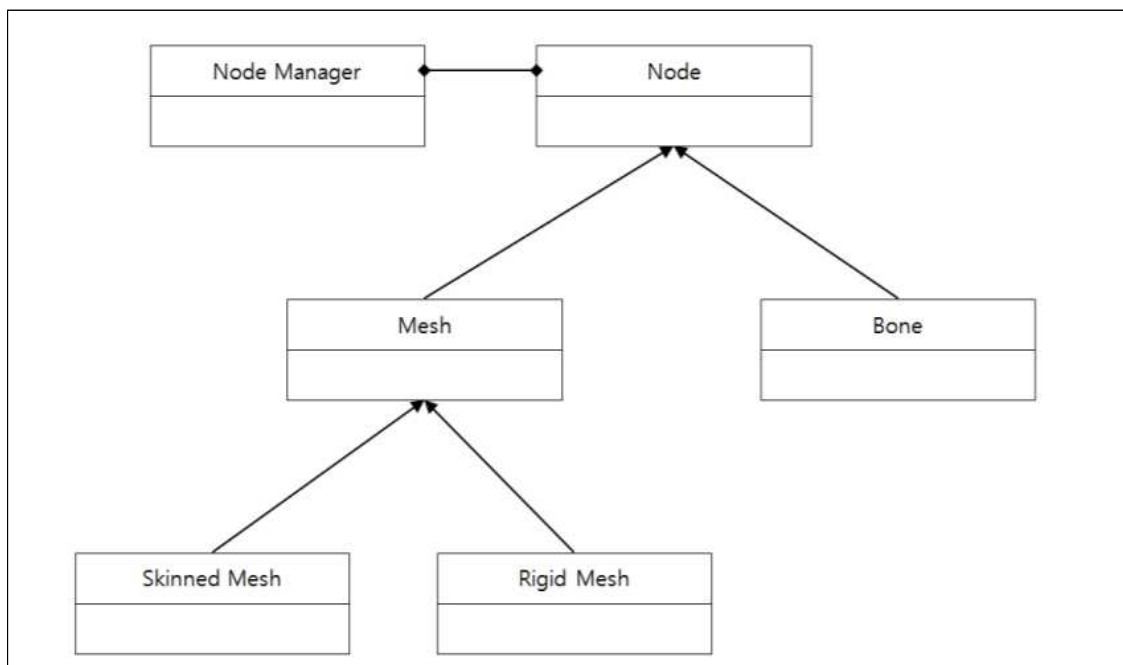
- 컴퓨터, VOOOOOO SOOOOOO 2019 이상 버전, Direct X SDK 9.0C 이상 버전

안전 · 유의 사항

- 컴퓨터 작업실에서 지켜야 할 안전 규칙을 준수한다.
- 정전이나 시스템 중대 오류로 발생하는 갑작스러운 컴퓨터 종료에 대비하여 소스 데이터를 백업한다.
- 프로그래밍 실습 시, C++ 언어의 포인터 자료형을 잘못 사용하여 시스템의 중대 오류가 발생하여 재부팅될 수 있으므로 실행 전에 소스 데이터를 반드시 저장한다.

수행 순서

- ① 캐릭터 애니메이션 시스템을 설계한다.



출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원.
[그림 6-1] 캐릭터 애니메이션 시스템 설계의 예

1. 노드(node)는 파일로부터 읽어 들인 모든 오브젝트(object)들의 최상위 클래스이다. 오브젝트(object)는 메시(mesh)와 본(bone)으로 구분하여 설정한다.
2. 스킨드 메시(skinnedMesh) 클래스는 스키닝 메시를 위한 클래스이며, 리지드 메시(rigidMesh)는 메시 클래스로부터 상속받은 클래스로, 정점 버퍼와 인덱스 버퍼를 생성하고 메시를 그려 주는 역할을 하도록 초기 설계를 한다.

② 노드 매니저(node manager) 클래스를 작성한다.

1. 모델링 데이터를 읽어 들여 실제로 화면에 출력 가능한 형태로 변환하는 역할을 하도록 기본 기능을 제작한다.
2. 노드 매니저(NodeMgr) 클래스는 오브젝트의 정보를 가지고 있는 오브젝트 인포(ObjectInfo)와 머티리얼(material) 정보를 가지고 있는 머티리얼들의 전역 데이터만을 읽어 들인 뒤에 각각의 메시 정보들이 어떤 클래스의 객체로 생성되어야 하는지를 판단하는 역할을 하도록 작성한다.

〈표 6-1〉 노드 매니저(node manager) 클래스 작성의 예

```

01  class NodeMgr
02  {
03      protected:
04          ObjectInfo      m_info;          // 현재 노드 전체에 대한 정보
05          vector<ZMaterial> m_materials;    // 재질 값 배열
06          vector<ZNode*>   m_nodes;        // 노드들
07          LPDIRECT3DDEVICE9 m_nodes;       // D3D 디바이스
08          D3DXMATRIXA16    m_matTM;       // 자식 노드 전체에 적용될 TM
09      public:
10          NodeMgr( LPDIRECT3DDEVICE9 pDev, CParsedData* pData);
11          ~NodeMgr( );
12          // TM을 얻어 온다.
13          D3DXMATRIXA16* GetTM( ) { &m_matTM; }
14          // TM을 세팅한다.
15          void SetTM(D3DXMATRIXA16* pTM ) { m_matTM = *pTM; }
16          // 애니메이션 행렬을 만든다.
17          int Animate( float fFrame );
18          // 노드 전체를 그린다.
19          int Draw( );
20  };

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

6-2. 플레이어 캐릭터 행동 구현

학습 목표

- 플레이어 캐릭터와 다른 캐릭터 및 게임 오브젝트 간의 상호 작용을 구현할 수 있다.
- 플레이어 캐릭터에 부가된 추가 아이템의 해제, 교체, 상호 작용을 구현할 수 있다.
- 플레이어 캐릭터 반응의 결과에 따른 보상과 캐릭터의 상태 정보를 구현할 수 있다.

필요 지식 /

① 뼈대 애니메이션(bone animation)

인간의 몸속에 뼈가 있듯이 3차원적으로 구성된 뼈대를 만들고, 이 뼈대에 메시를 자식으로 붙이는 방식이다. 이때, 기본적인 뼈대가 존재하는데 이러한 뼈대는 메시가 아니기 때문에 화면에 출력되지 않는다. 관절과 관절 사이를 융합하지 못하는 단점이 있지만, 동일한 뼈대 구조를 가진 오브젝트는 메시만 바꾸는 것으로도 똑같은 애니메이션이 가능하기 때문에 메모리의 절약과 다양한 응용이 가능하다는 장점이 있다.

② 스킨링 애니메이션(skinning animation)

스킨링(skinning)이란, 계층적 애니메이션과 뼈대 애니메이션 방식의 단점인 관절 부위의 문제를 해결하기 위한 방식이다. 스킨링을 구현하는 방법은 매우 다양하지만 주로 뼈대의 가중치(weight) 기법이 가장 많이 사용된다. 메시를 구성하는 각 정점이 뼈대로부터 힘을 얼마나 받는가 하는 가중치를 포함하고 있다.

1. 루프 연산

직접적으로 루프(loop)를 돌면서 정점, 가중치, 행렬을 연산하는 방법이 있다. 이론에 입각한 가장 정석적인 방법이지만, 하드웨어 가속 지원을 전혀 받지 못한다. 그 대신, 어떤 시스템에서도 확실하게 작동한다는 장점이 있다.

2. 매트릭스 팔레트(matrix palette)

매트릭스 팔레트(matrix palette)를 사용하는 방법이다. 가중치를 사용하는 정점을 생성하고, 256개의 매트릭스 팔레트에 애니메이션 매트릭스를 적절히 입력한 뒤 메시를 그리기만 하면 가속기가 가중치를 연산한다.

3. 정점 셰이더(vertex shader)

정점 셰이더(vertex shader)는 고급 셰이딩(shading) 언어로 그래픽 카드의 GPU에 프로그램을 작성하여 넣는 것이다. 복잡한 가중치 연산을 정점 셰이더와 GPU의 가속 기능을 통해 해결하는 것이므로, 사용자의 GPU가 정점 셰이더 가속을 지원하기만 하면 상당한 가속 효과를 볼 수 있다.

수행 내용 / 플레이어 캐릭터 행동 구현하기

재료·자료

- 시스템 기획서, 뼈대 애니메이션(bone animation)이 적용된 캐릭터 모델링(character modeling) 데이터, 모델링(modeling)에 입힐 텍스처(texture) 데이터

기기(장비·공구)

- 컴퓨터, VOOOOOO SOOOOOO 2019 이상 버전, Direct X SDK 9.0C 이상 버전

안전·유의 사항

- 컴퓨터 작업실에서 지켜야 할 안전 규칙을 준수한다.
- 정전이나 시스템 중대 오류로 컴퓨터가 갑자기 종료되는 사고에 대비하여 소스 데이터를 백업한다.
- 프로그래밍 실습 시, C++ 언어의 포인터 자료형을 잘못 사용하여 시스템의 중대 오류가 발생하면 컴퓨터가 재부팅될 수 있으므로 실행 전에 소스 데이터를 반드시 저장한다.

수행 순서

① 노드(node) 클래스를 작성한다. [그림 6-3] 참조.

1. 노드 클래스는 최상위 클래스이기 때문에 하위의 다양한 클래스를 관리할 수 있는 기능들이 추가되어야 한다.
2. 노드 클래스는 애니메이션에 관련 정보를 포함시키고, 그 정보들을 관리하도록 작성한다.

② 메시(mesh) 클래스를 작성한다. [그림 6-4] 참조.

1. 메시 클래스는 노드 클래스를 상속받아 정점 버퍼와 인덱스 버퍼의 데이터 구조를 선언한다.
2. 메시 클래스는 스킨드 메시(skinnedMesh)와 리지드 메시(rigidMesh)를 위한 인터페이스와 기본 멤버 변수를 선언하는 것을 주된 역할로 한다.
3. 정점 버퍼와 인덱스 버퍼의 선언은 메시 클래스 내에서 이루어지지만, 실제로 데이터를 담을 버퍼 공간은 스킨드 메시(skinnedMesh)와 리지드 메시(rigidMesh)에서 생성되는 구조로 설계한다.

③ 본(bone) 클래스를 작성한다. [그림 6-5] 참조.

1. 본 클래스는 노드 클래스로부터 상속받아 생성된 클래스로, 인터페이스만 제공하는 역할을 한다.
2. 실제로 메시를 그리기 위한 가상 함수를 포함하여 인터페이스 형태로 작성한다.

④ 리지드 메시(rigidMesh) 클래스를 작성한다. [그림 6-6] 참조.

리지드 메시(rigidMesh) 클래스는 메시 클래스를 상속받아 생성된 클래스로, 'm_vtx'와 'm_idx'에 값을 읽어 들이는 역할을 수행한다. 그리고 이 값들을 바탕으로 정점 버퍼와 인덱스 버퍼를 생성한다.

'Draw()' 함수에서는 노드의 변환 행렬(m_matTM)을 세팅하고, 메시의 정점 버퍼(vertex buffer)와 인덱스 버퍼(index buffer)를 가지고 'DrawIndexedPrimitive()' 함수로 메시지를 그리는 역할을 수행하도록 작성한다.

〈표 6-2〉노드(node) 클래스 작성의 예

```

01 class Node
02 {
03     protected:
04         enum NODETYPE { NODE_NODE, NODE_BONE, NODE_MESH };
05         // 현재 지원하는 노드의 종류
06         int m_nNodeType; // 노드의 종류
07         int m_nObjectID; // 오브젝트 ID
08         int m_nParentID; // 부모의 ID
09         int m_nMaterialID; // 재질의 ID
10         int m_matLocal; // 로컬 TM 행렬
11         int m_matAni; // 애니메이션 행렬
12         D3DXMATRIXA16 m_matTM; // 최종 TM 행렬
13         Track* m_pTrack; // 애니메이션 트랙 정보
14         LPDIRECT3DDEVICE9 m_pDev; //D3D 디바이스
15     public:
16         Node(LPDIRECT3DDEVICE9 pDev, CMesh* pMesh );
17         ~Node( );
18         int GetObjectID( ) { return m_nObjectID; } // 메시의 ID 값을 얻는다.
19         int GetParentID( ) { return m_nParentID; } // 부모 메시의 ID값을 얻는다.
20         int GetMaterialID( ) { return m_nMaterialID; }
21         D3DXMATRIXA16* GetMatrixTM( ) { return &m_matTM; }
22         D3DXMATRIXA16* Animate(float fFrame, D3DXMATRIXA16* pParentTM);
23         // Track으로부터 m_matAni를 구해서 m_matTM을 만든다.
24         Virtual int Draw( ) { return 1; } // 노드를 그린다.
25 };

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원.

〈그림 6-3〉 메시 클래스 작성의 예

```

01 class Mesh: public Node
02 {
03     protected:
04         BOOL            m_isSkinned; // 스키닝이 적용되는 메시
05         DWORD           m_nVerts;    // 전체 정점의 개수
06         DWORD           m_nTriangles; // 그릴 삼각형의 개수
07         DWORD           m_dwFVF;     // 정점의 FVF 값
08         LPDIRECT3DVERTEXBUFFER9 m_pVB; // 정점 버퍼
09         LPDIRECT3DINDEXBUFFER9  m_pIB; // 인덱스 버퍼
10     protected:
11         virtual int CreateVIB(CMesh* pMesh) { return 1; }
12     public:
13         Mesh(LPDIRECT3DDEVICE9 pDev, CMesh* pMesh);
14         ~Mesh( );
15         // 메시지를 그린다.
16         virtual int Draw( );
17 };

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원.

〈표 6-4〉 메시 클래스 작성의 예

```

01 class Bone: public Node
02 {
03     public:
04         Bone(LPDIRECT3DDEVICE9 pDev, CMesh* pMesh);
05         ~Bone( );
06
07         // 메시지를 그린다
08         virtual int Draw( );
09 };

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원.

〈표 6-5〉 리지드 메시(rigidMesh) 클래스 작성의 예

```

01 class RigidMesh: public Node
02 {
03     private:
04         // 정점 버퍼와 똑같은 점점 배열을 복사해 보관해 둔다
05         vector<RigidVertex>    m_vtx;    // 스키닝 없는 정점 배열
06         // 인덱스 버퍼와 똑같은 인덱스 배열을 복사해 보관해 둔다
07         vector<Index3w>        m_idx;    // 인덱스 배열
08         int CreateVIB(CMesh* pMesh);
09     public:
10         RigidMesh(LPDIRECT3DDEVICE9 pDev, CMesh* pMesh);
11         ~RigidMesh( );
12         int Draw( );
13 };

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원.

〈표 6-6〉 노드매니저(node manager) 클래스 작성의 예

```

01 // 인덱스가 32비트 인덱스를 지원하면 32비트 인덱스 버퍼 생성
02 if( caps.MaxVertexIndex > 0x0000ffff )
03 {
04     m_pDev->CreateIndexBuffer( m_nTriangles * sizeof(Index3i), 0
                                , D3DFMT_INDEX32, D3DPOOL_DEFAULT, &m_pIB, NULL
05 );
06     m_pIB->Lock( 0, m_nTriangles*sizeof(Index3i),(void**)&pl, 0 );
07     DWORD* pW =(DWORD*)pl;
08     for( i = 0 ; i < m_nTriangles ; i++ )
09     {
10         *(pW+0) = pMesh->m_idxFinal[i].i[0];
11         *(pW+1) = pMesh->m_idxFinal[i].i[1];
12         *(pW+2) = pMesh->m_idxFinal[i].i[2];
13         pW += 3;
14     }
15     m_pIB->Unlock( );
16 }
17 else // 아니라면 16비트 인덱스로 인덱스 버퍼 생성
18 {
19     m_pDev->CreateIndexBuffer( m_nTriangles * sizeof(Index3w), 0
                                , D3DFMT_INDEX16, D3DPOOL_DEFAULT, &m_pIB, NULL );
20
21     m_pIB->Lock( 0, m_nTriangles*sizeof(Index3w),(void**)&pl, 0 );
22     WORD* pW =(WORD*)pl;
23     for( i = 0 ; i < m_nTriangles ; i++ )
24     {
25         *(pW+0) =(WORD)pMesh->m_idxFinal[i].i[0];
26         *(pW+1) =(WORD)pMesh->m_idxFinal[i].i[1];
27         *(pW+2) =(WORD)pMesh->m_idxFinal[i].i[2];
28         pW += 3;
29     }
30     m_pIB->Unlock( );
31 }

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.

수행 tip

- 캐릭터 애니메이션 클래스는 복잡한 클래스의 집합체이다. 각 클래스의 역할에 대한 명확한 이해를 바탕으로 실습하도록 지도한다.

교수 방법

- 뼈대(bone)와 메시(mesh)의 개념과 차이에 대한 지식을 충분히 전달하고 이해시킨 다음 단계적 실습을 지도한다.
- 여러 가지 입출력 데이터에 따른 캐릭터 이동을 표현하기 위한 기법을 예시로 제시하고, 장단점과 동작 구조의 차이점을 이해할 수 있도록 지도한다.
- 플레이어 캐릭터의 다양한 상호 작용을 구현할 수 있도록 지도한다.

학습 방법

- 각 노드를 설계할 때 메시와 뼈대의 개념을 명확히 이해하고 설계한다.
- 예제 프로그램으로 충분히 사용법을 익힌 다음 설계 내용을 구현한다.
- 입출력 데이터에 따른 플레이어 캐릭터의 행동을 구현한다.

평가 준거

- 평가자는 학습자가 학습 목표를 성공적으로 달성하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가해야 한다.

학습 내용	학습 목표	성취수준		
		상	중	하
입출력 데이터에 따른 캐릭터 반응 설계 및 구현	- 사용자가 조작 가능한 입출력 및 캐릭터 반응을 설계할 수 있다.			
	- 사용자 입출력에 반응한 캐릭터의 동작과 반응을 구현할 수 있다.			
플레이어 캐릭터 행동 구현	- 플레이어 캐릭터와 다른 캐릭터 및 게임 오브젝트 간의 상호 작용을 구현할 수 있다.			
	- 플레이어 캐릭터에 부가된 추가 아이템의 해제, 교체, 상호 작용을 구현할 수 있다.			
	- 플레이어 캐릭터 반응의 결과에 따른 보상과 캐릭터의 상태 정보를 구현할 수 있다.			

평가 방법

- 포트폴리오

학습 내용	평가 항목	성취수준		
		상	중	하
입출력 데이터에 따른 캐릭터 반응 설계 및 구현	- 입출력 프로그램 시스템의 구성 요소를 명확하게 구현하는 능력			
	- 캐릭터 애니메이션의 원리를 이해하고 프로그래밍에 효과적으로 적용하는 능력			
플레이어 캐릭터 행동 구현	- 코드를 명확하게 작성하고, 정확한 데이터를 도출하는 능력			
	- 테스트를 효율적으로 수행하고, 디버깅을 능숙하게 수행하는 능력			
	- 작성된 프로그래밍 코드에 설명을 첨부하고, 별도로 문서화를 수행했는지 여부			

• 서술형 시험

학습 내용	평가 항목	성취수준		
		상	중	하
입출력 데이터에 따른 캐릭터 반응 설계 및 구현	- 입출력에 따른 캐릭터 동작 원리에 대해 명확하게 이해하고 있는지 여부			
	- 캐릭터 애니메이션을 구현하기 위한 다양한 구조적 요소에 대해 이해하고 있는지 여부			
플레이어 캐릭터 행동 구현	- 캐릭터 프로그램을 구성하고 있는 다양한 구조적 요소의 역할을 이해하고 있는지 여부			
	- 캐릭터 프로그램을 구현하기 위한 충분한 수학적 지식을 이해하고 있는지 여부			

• 평가자 체크리스트

학습 내용	평가 항목	성취수준		
		상	중	하
입출력 데이터에 따른 캐릭터 반응 설계 및 구현	- 캐릭터 동작 프로그램 구현 알고리즘에 대한 이해 여부			
	- 입출력 데이터 프로그램을 위한 구성 요소의 이해 여부			
플레이어 캐릭터 행동 구현	- 애니메이션 조작 프로그램을 위한 수학적 지식의 이해 여부			
	- 플레이어 캐릭터의 상호 작용에 따른 상태 정보 설계 능력			

피드백

1. 포트폴리오

- 뼈대와 메시가 연결된 상태, 애니메이션 시 메시가 이탈되는 현상 등을 각 설계 문서대로 적절히 구현하고 있는지 체크리스트로 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 주요 사항을 지적하여 설명해 준다.

2. 서술형 시험

- 캐릭터 이동 프로그램의 원리에 대해 평가한 뒤, 평가 결과가 미흡한 학습자에게는 다시 설명한다.
- 입출력에 따른 애니메이션 재생 순서의 설계에 대해 평가한 뒤, 평가 결과가 우수한 학습자에 대해서는 기능을 구현할 수 있도록 설명해 준다.

3. 평가자 체크리스트

- 뼈대와 뼈대가 체결된 부분과 동작 구현 시 부자연스러운 부분을 체크리스트로 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 주요 사항을 표시하여 설명해 준다.
- 뼈대와 메시의 연결 부분과 메시 간 연결 동작이 부자연스러운 부분을 비교하여 체크리스트로 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 주요 사항을 표시하여 설명해 준다.

학습 1	게임 서버 시스템 설계하기
학습 2	서버 네트워크 프로그래밍하기
학습 3	C/S 모델에서의 클라이언트 네트워크 프로그래밍하기
학습 4	P2P 모델의 클라이언트 네트워크 프로그래밍하기
학습 5	게임 서버 응용 프로그래밍하기
학습 6	플레이(캐릭터) 프로그래밍하기
학습 7	지형 프로그래밍하기
학습 8	이벤트 프로그래밍하기

7-1. 지형과 월드 설계 및 구현

학습 목표

- 게임 기획에 따른 게임의 지형과 월드를 설계할 수 있다.
- 게임 플레이에 적합한 인스턴스 던전을 구현할 수 있다.

필요 지식 /

① 높이 맵

등고선에서는 높이 값을 등고선의 색깔 값으로 나타낸다. 높이 맵은 높이 값을 0~255 사이의 명암 값으로 나타낸 것이라고 할 수 있다.

1. 정의

높이 값만으로 지형을 구성하는 것을 '높이 맵(height field)'이라고 한다. 위에서 수직으로 보았을 때 x와 z축에 대한 격자가 동일하여 높이에 대한 정보만 가지고 있으면 지형을 표현할 수 있기 때문에 '높이 맵'이라고 부른다.

2. 특징

(1) 높이 값

높이 값만 가지고 있으므로 데이터의 크기가 작고, 높이 값은 포OO과 같은 그래픽 툴로 흑백 이미지를 만들 수 있다는 장점이 있다. 따라서 별도의 툴이 없더라도 지형을 손쉽게 만들 수 있다. 또한, 거대한 지형을 블록 단위로 그리면 같은 인덱스 버퍼를 사용할 수 있기 때문에 메모리를 절약할 수 있다. 이 밖에 오브젝트가 지형 위에 있을 때 산술 연산만으로 높이를 구할 수 있으며, 멀티 텍스처를 활용한 타일링(tiling) 등의 효과를 표현하기가 쉽다.

(2) 기울기

기울기가 급한 지형일수록 메시지를 더 많이 넣어 부드럽게 표현해 주어야 하는데 높이 맵은 이것이 불가능하며, 지형의 기울기가 작으면 메시가 많이 필요하지 않음에도 불구하고 메시의 정점을 줄이지 못한다는 단점이 있다. 이러한 단점을 극복하기 위해 쿼드 트리(quad tree), ROAM 등이 개발되었는데, 이 중에서 ROAM은 실외 지형에 대한 메시의 표현력이 가장 우수하여 비행 시뮬레이션과 같은 게임에서 지형을 표현하는 데 많이 사용된다.

수행 내용 / 지형과 월드 설계와 구현하기

재료·자료

- 시스템 기획서

기기(장비 · 도구)

- 컴퓨터
- VOOOOO SOOOOO 2019 이상 버전
- Direct X SDK 9.0C 이상 버전

안전 · 유의 사항

- PC 작업실에서 지켜야 할 안전 규칙을 준수하여 실습에 임한다.
- 정전이나 시스템 중대 오류로 컴퓨터 갑자기 종료되는 사고에 대비하여 소스 데이터를 백업한다.
- 프로그래밍 실습 시, C++ 언어의 포인터 자료형을 잘못 사용하여 시스템에 중대 오류가 발생하면 재부팅될 수 있으므로 실행 전에 소스 데이터를 반드시 저장한다.

수행 순서

- ① 높이 맵을 제작 실습한다. [그림 7-1] 참조.



```

CUSTOMVERTEX v;
CUSTOMVERTEX*pV =(CUSTOMVERTEX*)pVertices;
    for( DWORD z = 0 ; z < g_czHeight ; z++ )
    {
        for( DWORD x = 0 ; x < g_cxHeight ; x++ )
        {
            /// 정점의 x 좌표(메시를 원점에 정렬)
            v.p.x = (float)x-g_cxHeight/2.0f;
            /// 정점의 z 좌표(메시를 원점에 정렬),
            /// z축이 모니터 안쪽이므로 -를 곱한다.
            v.p.z = -((float)z-g_czHeight/2.0f);
            /// DWORD이므로 pitch/4
            v.p.y = (float)((LPDWORD)d3drc.pBits + x + z
                            *(d3drc.Pitch/4)) & 0x000000ff)/10.0f;
            v.n.x = v.p.x;
            v.n.y = v.p.y;
            v.n.z = v.p.z;
            D3DXVec3Normalize( &v.n, &v.n );
            v.t.x =(float)x /(g_cxHeight-1);
            v.t.y =(float)z /(g_czHeight-1);
            *pV++ = v;
        }
    }

```

출처: 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534).
한국직업능력개발원.
[그림 7-1] 맵 제작 로직

7-2. 지형 내비게이션 알고리즘 구현

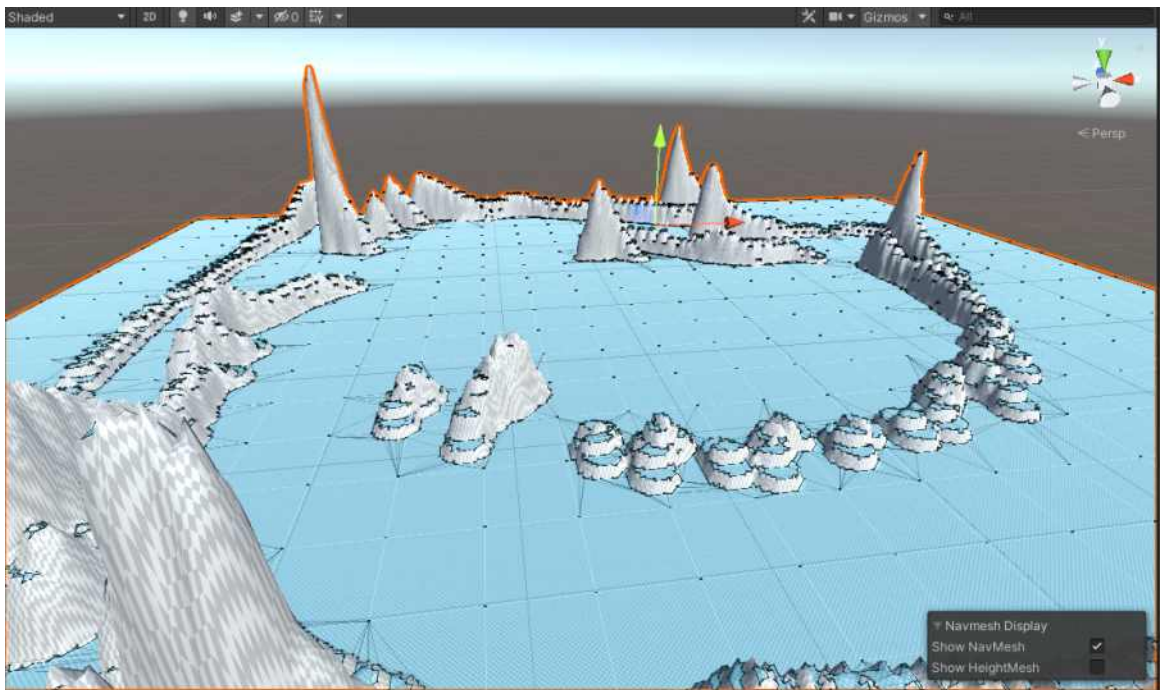
학습 목표

- 캐릭터의 오픈 플레이에 적합한 오픈 월드를 구현할 수 있다.
- 캐릭터 플레이에 적합한 물리적 반응, 지형 내비게이션, 지형 오브젝트의 자료 구조와 알고리즘을 구현할 수 있다.

필요 지식 /

① 내비 메시(NavMesh)

내비게이션 메시(navigation mesh)의 줄임말로, 게임 월드에서 걸을 수 있는 표면을 뜻하며, 내비 메시지를 사용하여 게임 월드의 한 위치에서 다른 위치로 이동할 수 있는 경로를 찾을 수 있다. 데이터 구조는 레벨 지오메트리(level geometry)에서 자동으로 빌드(build) 또는 베이크(bake)된다.



출처: 집필진 제작(2022)
[그림 7-2] 지형 네비 메시

수행 내용 / 지형 내비게이션 알고리즘 구현하기

재료·자료

- 시스템 기획서

기기(장비·공구)

- 컴퓨터, VOOOOO SOOOOO 2019 이상 버전, Direct X SDK 9.0C 이상 버전

안전·유의 사항

- PC 작업실에서 지켜야 할 안전 규칙을 준수하여 실습에 임한다.
- 정전이나 시스템 중대 오류로 갑작스러운 컴퓨터 종료에 대비하여 소스 데이터를 백업한다.
- 프로그래밍 실습 시, C++ 언어의 포인터 자료형을 잘못 사용하여 시스템에 중대 오류가 발생하면 재부팅될 수 있으므로 실행 전에 소스 데이터를 반드시 저장한다.

수행 순서

1. 정점에 위치 값과 색상 값을 적용한다.

정점에 디퓨즈(diffuse) 값과 텍스처를 적용하기 위해서는 가장 먼저 정점 데이터에 색상 값을 적용할 수 있도록 [그림 7-3]과 같이 구조체와 FVF 값을 바꾸어야 한다.

- (1) 만약, 지형의 양면을 표현하려면 렌더링 상태 값인 컬링 모드(culling mode)를 양면을 그릴 수 있는 'None'으로 설정한다.
- (2) 'm_pDev → SetRenderState(D3DRS_CULLMODE, D3DCULL_NONE); //'는 컬링 모드(culling mode)를 지정하지 않는다는 코드이다.

```
struct VtxD
{
    D3DXVECTOR3 p;    // position
    DWORD d;         // diffuse color
    ...
    enum {FVF =(D3DFVF_XYZ|D3DFVF_DIFFUSE),}; // FVF
};
```

2. 정점에 텍스처 UV 좌표값을 추가한다.

- (1) 디테일 맵(detail map)의 크기는 작고, 처리 방법 또한 간편하여 3D 게임 초창기 비디오 메모리가 크지 않았을 때에는 단조로운 지형의 색상에 효과적이었다. 현재에도 시스템 메모리가

PC보다 작은 임베디드(embedded) 기기에 일부 적용되기도 한다.

디테일 맵은 정점에 텍스처를 입히는 것과 같기 때문에 텍스처의 UV 좌표가 설정될 수 있도록 [그림 7-4]와 같이 정점 구조체와 FVF를 구성한다.

```
struct VtxDUV1
{
    D3DXVECTOR3 p;    // position
    DWORD d;          // diffuse
    FLOAT u,v;        // texture u,v
    . . .
    enum {FVF =(D3DFVF_XYZ|D3DFVF_DIFFUSE|D3DFVF_TEX1),};
};
```

- (2) 지형 전체에 [0, 1] 범위로 텍스처를 적용하려면, 정점의 U, V를 타일의 숫자로 나눈다. UV 설정 시 주의해야 할 것은 3D 지형에 적용되는 텍스처들이 대부분 z축으로 증가하면, UV 좌표의 v 값을 감소시키기 위해서 RAW 파일을 지형의 높이 적용에서 그림의 위아래 데이터를 바꾸었듯이, UV의 v 값도 위아래를 바꾸어야 한다는 것이다. 바꿀 때에는 '1.0 - v'로 설정한다.

```
m_pVtx[n].u = FLOAT(x)/m_TileN;
m_pVtx[n].v = 1.f - FLOAT(z)/m_TileN;
```

- (3) 디테일 맵(detail map)의 목적은 노이즈(noise) 등을 이용하여 질감의 자연스러움을 표현하는 것인데, UV의 [0, 1] 범위 값은 적당하지 않음을 바로 알 수 있다. 계산된 UV의 [0, 1] 범위 값은 이후에도 이용될 수 있으므로 디테일 맵의 반복 적용에 대한 적당한 셀 범위의 변수 하나를 추가한 뒤 UV에 곱해 사용한다. 이 실습 사례에서는 16.0f 정도의 값을 사용한다.

```
m_fUV = 16.f;
m_pVtx[n].u = FLOAT(x)/m_TileN;
m_pVtx[n].v = 1.f - FLOAT(z)/m_TileN;
m_pVtx[n].u *= m_fUV; // 디테일 텍스처의 u 값을 증가시킴
m_pVtx[n].v *= m_fUV; // 디테일 텍스처의 v 값을 증가시킴
```

3. 정점에 다중 텍스처 처리(multi-texturing) UV 좌표값을 추가한다.

- (1) 정점의 색상, 디퓨즈 맵, 디테일 맵을 적용하는 다중 텍스처 처리(multi-texturing)을 위해 2개의 텍스처 좌표가 사용될 수 있게 정점 구조체와 FVF를 변경한다.

여기에서 정점 데이터의 u0, v0는 디퓨즈 맵의 좌표로 사용하고, u1, v1은 디테일 맵의 좌표로 사용한다.

```

struct VtxDUV2
{
    D3DXVECTOR3 p; // position
    DWORD d; // diffuse
    FLOAT u0, v0; // diffuse map uv
    FLOAT u1, v1; // detail map uv
    . . .
    enum {FVF =(D3DFVF_XYZ|D3DFVF_DIFFUSE|D3DFVF_TEX2),};
};

```

(2) 디퓨즈 맵과 디테일 맵의 혼합은 곱셈 연산으로 처리한다.

고정 기능 파이프라인(pipeline)에서 곱셈 연산에 대한 OP는 MODULATE, MODULATE2X, MODULATE4X가 있다.

```

// 0-stage color operation
m_pDev->SetTextureStageState(0, D3DTSS_COLORARG1, D3DTA_TEXTURE);
m_pDev->SetTextureStageState(0, D3DTSS_COLORARG2, D3DTA_DIFFUSE);
m_pDev->SetTextureStageState(0, D3DTSS_COLOROP, D3DTOP_MODULATE);
// 0-stage color operation
m_pDev->SetTextureStageState(0, D3DTSS_ALPHAARG1, D3DTA_TEXTURE);
m_pDev->SetTextureStageState(0, D3DTSS_ALPHAARG2, D3DTA_DIFFUSE);
m_pDev->SetTextureStageState(0, D3DTSS_ALPHAOP, D3DTOP_MODULATE);
// 1-stage alpha operation
m_pDev->SetTextureStageState(1, D3DTSS_COLORARG1, D3DTA_CURRENT);
m_pDev->SetTextureStageState(1, D3DTSS_COLORARG2, D3DTA_TEXTURE);
m_pDev->SetTextureStageState(1, D3DTSS_COLOROP, D3DTOP_MODULATE2X);
// 1-stage alpha operation
m_pDev->SetTextureStageState(1, D3DTSS_ALPHAARG1, D3DTA_CURRENT);
m_pDev->SetTextureStageState(1, D3DTSS_ALPHAARG2, D3DTA_TEXTURE);
m_pDev->SetTextureStageState(1, D3DTSS_ALPHAOP, D3DTOP_MODULATE);

```

④ 작성된 높이 맵 함수의 테스트 및 디버깅 다중 텍스처 처리(multi-texturing)는 텍스처의 레이어 수에 따라 렌더링 성능이 급격히 저하된다. 따라서 한 지형 데이터에 사용될 텍스처의 수를 적정선으로 제한하여 사용한다.

⑤ 완성된 높이 맵 함수를 이용하여 게임 기획서에 맞는 지형 데이터를 만든다. 다중 텍스처 처리는 텍스처의 레이어 수에 따라 렌더링 성능이 급격히 저하된다. 따라서 한 지형 데이터에 사용될 텍스처의 수를 적정선으로 제한하여 사용한다.

수행 tip

- 곱셈 연산을 계속하면 색상은 점점 어두워지므로 먼저 MODULATE로 처리해 보고, 어둡다고 생각되면 MODULATE2X나 MODULATE4X를 적용하여 색상의 밝기를 올리고, 이 중에서 가장 알맞은 OP를 선택한다.

교수 방법

- 3D 그래픽 용어와 툴 사용법을 충분히 전달하고 이해시킨 다음 단계적 실습을 지도한다.
- 높이 맵 제작 시 오브젝트가 맵에 위치할 때 이질감을 줄이기 위해 고려해야 할 사항을 미리 숙지할 수 있도록 지도한다.
- 지형 데이터와 지형 텍스처 제작 시 놓치기 쉬운 리소스와 텍스처 간의 비조화에 대해 이해시키고 숙지할 수 있도록 지도한다.

학습 방법

- 3D 프로그램 툴 사용법과 관련 용어들을 충분히 숙지한 다음 단계적 실습을 시작한다.
- 높이 맵 제작 시 오브젝트가 놓여질 부분을 항상 체크하여 이질감을 줄인다.
- 예제 프로그램으로 3D 프로그램 툴의 사용법을 충분히 숙지한 뒤 기획 내용을 구현한다.

평가 준거

- 평가자는 학습자가 학습 목표를 성공적으로 달성하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가해야 한다.

학습 내용	학습 목표	성취수준		
		상	중	하
지형과 월드 설계 및 구현	- 게임 기획에 따른 게임의 지형과 월드를 설계할 수 있다.			
	- 게임 플레이에 적합한 인스턴스 던전을 구현할 수 있다.			
지형 내비게이션 알고리즘 구현	- 캐릭터의 오픈 플레이에 적합한 오픈 월드를 구현할 수 있다.			
	- 캐릭터 플레이에 적합한 물리적 반응, 지형 내비게이션, 지형 오브젝트의 자료 구조와 알고리즘을 구현할 수 있다.			

평가 방법

- 포트폴리오

학습 내용	평가 항목	성취수준		
		상	중	하
지형과 월드 설계 및 구현	- 높이 맵을 이해하고 있는지 여부			
	- 정점 버퍼를 조작하여 의도하는 지형 데이터를 작성하는 능력			
	- 지형 데이터에 원하는 텍스처를 자유롭게 적용하는 능력			
지형 내비게이션 알고리즘 구현	- 다중 텍스처 처리(Multi-texturing)를 자유롭게 적용하는 능력			
	- 테스트를 효율적으로 수행하고, 디버깅을 능숙하게 작업하는 능력			
	- 작성된 프로그래밍 코드에 설명을 첨부하였으며, 별도로 문서화를 수행하였는지 여부			

• 서술형 시험

학습 내용	평가 항목	성취수준		
		상	중	하
지형과 월드 설계 및 구현	- 높이 맵의 기본 원리에 대한 이해 여부			
	- 높이 맵을 표현하기 위한 그래픽 프로그래밍 절차에 대한 이해 여부			
지형 내비게이션 알고리즘 구현	- 다중 텍스처 처리(Multi-texturing)의 개념과 원리에 대한 이해 여부			

• 평가자 체크리스트

학습 내용	평가 항목	성취수준		
		상	중	하
지형과 월드 설계 및 구현	- 높이 맵에 대한 원리를 이해하고 있는지 여부			
	- 높이 맵을 구현하기 위한 로직을 이해하고 있는지 여부			
지형 내비게이션 알고리즘 구현	- 높이 맵에 텍스처를 적용하기 위한 방법과 알고리즘을 이해하고 있는지 여부			
	- 다중 텍스처 처리(Multi-texturing)와 적용 방법을 이해하고 있는지 여부			

피드백

1. 포트폴리오

- 기획서에서 요구한 지형 데이터, 맵, 텍스처 등을 각 설계 문서대로 적절히 구현하였는지 체크리스트로 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 주요 사항을 지적하여 설명해 준다.

2. 서술형 시험

- 높이 맵의 기본 원리에 대하여 평가한 뒤, 평가 결과가 미흡한 학습자에게는 다시 설명해 준다.
- 지형과 월드 설계에 대해서 평가한 뒤, 충분히 이해한 학습자에 대해서는 내비게이션 알고리즘에 대해서 설명해 준다.

3. 평가자 체크리스트

- 높이 맵에 대한 이해와 구현된 로직을 체크리스트로 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 주요 사항을 표시하여 설명해 준다.
- 지형 데이터에 텍스처가 설계한 문서의 내용과 비교하여 체크리스트로 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 주요 사항을 표시하여 설명해 준다.

학습 1	게임 서버 시스템 설계하기
학습 2	서버 네트워크 프로그래밍하기
학습 3	C/S 모델에서의 클라이언트 네트워크 프로그래밍하기
학습 4	P2P 모델의 클라이언트 네트워크 프로그래밍하기
학습 5	게임 서버 응용 프로그래밍하기
학습 6	플레이(캐릭터) 프로그래밍하기
학습 7	지형 프로그래밍하기
학습 8	이벤트 프로그래밍하기

8-1. 게임 플레이 이벤트 설계 및 구현

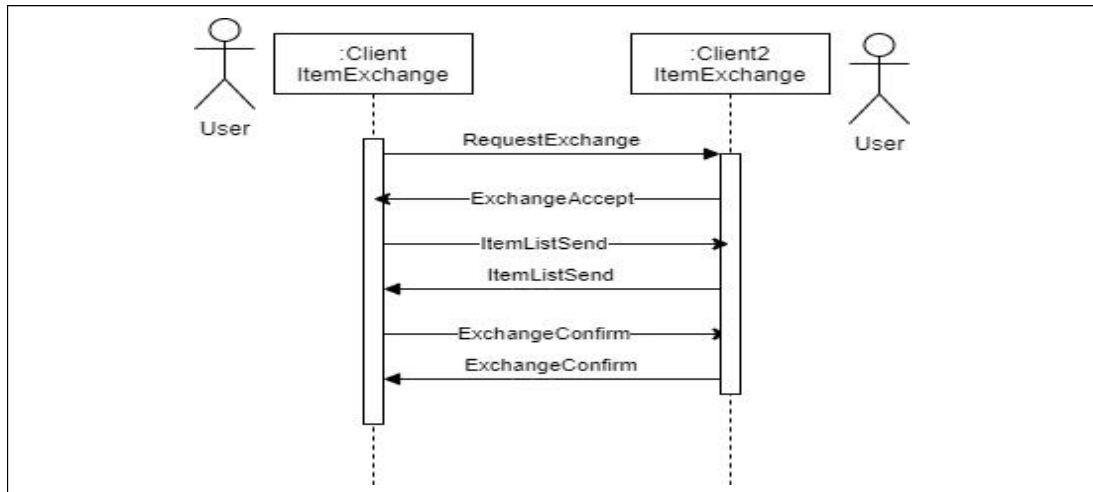
학습 목표

- 게임 기획에 따른 게임 플레이 이벤트 설계를 할 수 있다.
- 캐릭터의 아이템 인벤토리, 아이템 교체, 아이템 효과 반영에 대한 이벤트를 구현할 수 있다.

필요 지식 /

① 캐릭터 간의 아이템 교환 거래

캐릭터 간의 거래는 [그림 8-1]과 같이 한 쪽에서 먼저 거래를 요청하고, 상대방은 거래 요청에 응할지를 생각하여 허락하면 아이템 거래가 시작된다. 거래자 양쪽은 교환할 아이템 리스트를 전송하여 아이템을 살펴보고, 거래를 원할 경우에 거래 확인 패킷을 전송한다. 한 쪽에서만 거래 확인 패킷을 전송하면 단순히 캐릭터의 거래 상태만을 변경시키지만, 이미 상대방이 거래 확인 패킷을 전송했을 경우에는 양쪽 모두 동의하는 것이므로 아이템 교환이 이루어진다.



출처: 집필진 제작(2022)

[그림 8-1] 유저 간의 아이템 교환 거래 순서

② 아이템 착용에 따른 기본 정보 변화

일반적인 온라인 게임에서 캐릭터가 장착하는 아이템은 캐릭터의 능력치를 올려 주는 속성을 가지고 있기 때문에 캐릭터가 아이템을 착용하거나 해제할 경우에 캐릭터의 능력치가 변한다. 그렇기 때문에, 아이템을 착용하거나 해제할 때에는 캐릭터의 능력치를 항상 다시 계산해야 한다.

수행 내용 / 게임 플레이 이벤트 설계와 구현하기

재료·자료

- 시스템 기획서

기기(장비·공구)

- 컴퓨터, VOOOOO SOOOOO 2019 이상 버전, Direct X SDK 9.0C 이상 버전

안전·유의 사항

- PC 작업실에서 지켜야 할 안전 규칙을 준수하여 실습에 임한다.
- 정전이나 시스템 중대 오류로 갑작스러운 컴퓨터 종료에 대비하여 소스 데이터를 백업한다.
- 프로그래밍 실습 시, C++ 언어의 포인터 자료형을 잘못 사용하여 시스템에 중대 오류가 발생하면 재부팅될 수 있으므로 실행 전에 소스 데이터를 반드시 저장한다.

수행 순서

① 게임 시스템 기획서에 맞게 플레이 이벤트 시스템을 설계한다.

② 사용자의 아이템 인벤토리(item inventory) 기능을 구현한다.

1. 아이템 타입을(item type)을 분류한다.

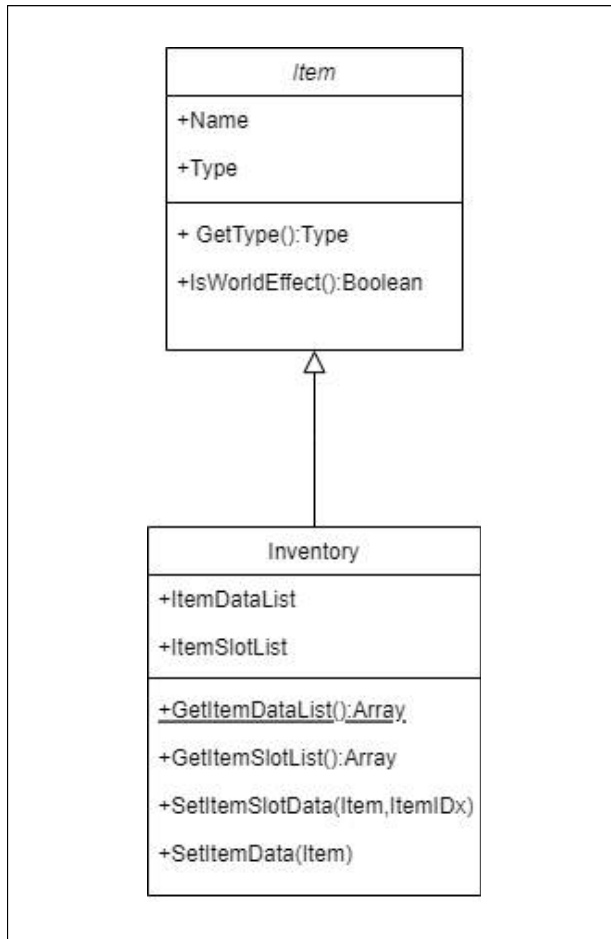
게임 시스템 기획서에 명시된 아이템 종류를 타입(type)화시킨다.

<i>Item</i>
+Name +Type
+ GetType():Type +IsWorldEffect():Boolean

출처: 집필진 제작(2022)
[그림 8-2] 아이템 구조

2. 인벤토리에 표시할 데이터를 구조화시킨다.

3. 구조화된 데이터를 기반으로 중첩 가능 여부에 따라서 아이템 배열을 구현한다.



출처: 집필진 제작(2022)

[그림 8-3] 인벤토리 클래스와 아이템 클래스 관계 다이어그램

③ 아이템 교체와 아이템 효과에 따른 데이터 표현 기능을 구현한다.

1. 아이템 교체를 위한 함수를 설계한다.
2. 아이템 교체 시 서버에 저장할 데이터를 구조화시킨다.
3. 아이템 교체를 위한 서버 통신 기능을 구현한다.
4. 아이템 사용 기능을 구현한다.
5. 아이템 사용이 월드에 적용되는지 여부를 체크한다.

아이템 효과 타입에 따른 월드 효과인지 로컬 효과인지 판단한다.

- (1) 월드 효과일 경우 서버 브로드캐스팅(broadcasting)으로 모든 유저에 적용되도록 구현한다.
- (2) 로컬(local) 효과일 경우 서버에 아이템 사용 데이터만 저장을 요청한다.

8-2. 상호 작용 이벤트 프로그래밍 구현

학습 목표

- 사용자 캐릭터와 타 사용자 및 비사용자 캐릭터 사이의 상호 작용에 대한 이벤트를 구현할 수 있다.
- 사용자 캐릭터와 지형 및 게임 오브젝트 사이의 상호 작용에 대한 이벤트를 구현할 수 있다.
- 사용자 캐릭터의 행동에 대한 보상 이벤트를 구현할 수 있다.

필요 지식 /

① 필드 서버 이동에서 서버와 클라이언트의 처리

캐릭터가 필드 서버를 이동할 때에는 가장 먼저 클라이언트가 캐릭터의 위치를 체크하고, 필드 이동 요청 패킷을 발송하여 서버를 이동할 것인지를 체크한다. 서버에서 체크할 수도 있지만, 서버의 부하를 줄이기 위해 클라이언트에서 체크하여 이동 요청 패킷을 서버로 전송한다. 서버에서 이동 요청 패킷을 수신하면 클라이언트가 송신한 패킷에서 이동 지역 인덱스를 추출하고, 캐릭터의 위치와 이동 지역 위치 사이의 거리를 계산하여 이동 요청의 유효성을 판별한다.

필드 이동에 따른 유저의 인증 작업은 접속 중인 서버에서 키를 생성하여 이동할 서버에 저장한 뒤 클라이언트에게 이동할 서버의 정보와 키 값을 전송한다. 클라이언트는 전송받은 서버 정보와 키 값을 이용하여 이동할 서버의 인증 작업을 수행하고, 성공하면 캐릭터는 필드 서버를 옮긴다.

② 데이터 저장 타이밍

1. 시간 간격에 따른 저장

시간 간격을 두고 저장하거나 유저가 접속을 끊을 때 저장하는 방법을 사용하면 데이터베이스에 질의하는 횟수가 줄어들기 때문에 데이터베이스의 부하를 줄일 수 있는 좋은 방법이지만, 서버가 안정적이어야 한다는 조건이 필요하다. 클로즈 베타테스트(closed beta test)나 오픈 베타테스트(open beta test) 등과 같이 서버의 안정성이 명확하게 확인되지 않은 시점에서 시간 간격으로 데이터를 저장하는 방법은 자칫 서버가 다운되어 유저의 데이터가 유실될 수 있는 위험 부담이 있기 때문이다.

2. 이벤트 발생에 따른 저장

이벤트가 발생할 때마다 해당 데이터를 저장하는 방법은 게임 서버가 다소 불안하여 다운되는 경우가 발생하더라도 다운 직전까지의 데이터는 모두 저장되므로, 유저의 데이터 유실을 최소화할 수 있다. 하지만 게임 서버에서는 많은 이벤트가 발생하므로 잦은 데이터베이스의 질의로 인해 데이터베이스 서버에 많은 부하를 줄 수 있다. 그러므로 게임 서버에는 고사양의 데이터베이스 시스템을 갖추어야 한다.

수행 내용 / 상호 작용 이벤트 프로그래밍 구현하기

재료·자료

- 시스템 기획서

기기(장비·공구)

- 컴퓨터, VOOOOO SOOOOO 2019 이상 버전, Direct X SDK 9.0C 이상 버전

안전·유의 사항

- PC 작업실에서 지켜야 할 안전 규칙을 준수하여 실습에 임한다.
- 정전이나 시스템 중대 오류로 갑작스러운 컴퓨터 종료에 대비하여 소스 데이터를 백업한다.
- 프로그래밍 실습 시, C++ 언어의 포인터 자료형을 잘못 사용하여 시스템에 중대 오류가 발생하여 재부팅될 수 있으므로 실행 전에 소스 데이터를 반드시 저장한다.

수행 순서

① 사용자 캐릭터 데이터 구조체를 만든다.

1. 게임 시스템 기획서에 있는 캐릭터 명세에 맞게 구조를 설계한다.
2. 설계된 구조를 통해 사용자 캐릭터 데이터를 관리하는 캐릭터 컨트롤러를 구현한다.

② 사용자 캐릭터 구조체를 배열 형태로 정의한다.

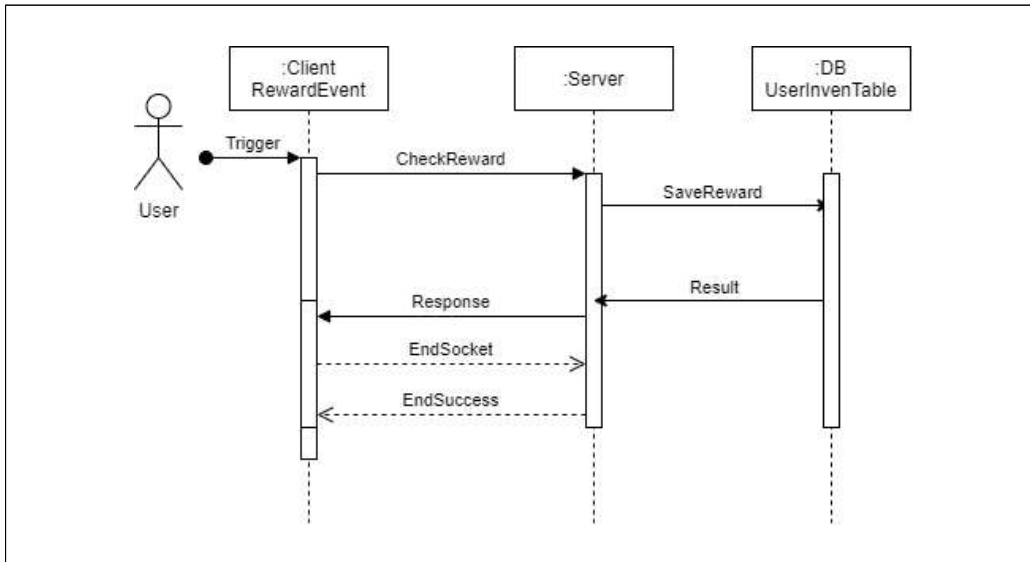
1. 사용자 캐릭터의 속성들을 확인한 뒤 데이터 구조를 설계한다.
2. 설계한 구조를 토대로 구조체를 작성한다.
3. 캐릭터 데이터 구조체를 담을 배열을 선언한다.

③ 사용자와 타 사용자의 상호 작용을 받을 수 있는 이벤트 함수를 구현한다.

④ 사용자 캐릭터와 지형 오브젝트의 상호 작용에 따른 이벤트 함수를 구현한다.

⑤ 캐릭터의 행동에 따른 보상 이벤트를 구현한다.

1. 보상 이벤트들의 명세서를 확인한다.
2. 명세에 따른 이벤트 처리 함수를 구현한다.
3. 보상 이벤트 요청에 따른 데이터 저장 기능을 구현한다.
4. 구현한 보상 이벤트 시스템을 테스트한 뒤 버그를 수정한다.



출처: 집필진 제작(2022)

[그림 8-4] 보상 이벤트 처리 시퀀스 다이어그램

수행 tip

- 완성도가 있는 예제 프로젝트를 공유한 뒤 게임 플레이 이벤트 시스템을 구현할 수 있도록 준비해야 하며, 조별 프로젝트로 실습한다.

교수 방법

- 게임 플레이 이벤트 설계의 이론적인 지식을 충분히 전달하고 이해시킨 다음 단계적인 실습을 지도한다.
- 이 학습 모듈은 팀 단위 프로젝트 중심으로 실습을 진행하도록 현장 지도한다.
- 학습 단위 주제별 실습 예제 코드를 유인물과 소스 코드 공유, 배포 등을 통해 진행할 수 있도록 사전에 준비한다.

학습 방법

- 이 실습 모듈은 멀티플레이 기능 구현을 목적으로 교육을 요구하고 있다. 따라서 통신을 위한 예제 프로젝트를 학습한 뒤, 이 학습 모듈의 실습을 한다.
- 게임 플레이에 따른 이벤트의 기능 및 흐름에 대해 파악한 뒤, 이벤트 프로그래밍 구조를 설계할 수 있도록 학습한다.
- 좀 더 완벽한 게임 서버 구축에는 많은 시간이 요구될 수 있다. 따라서 학습의 효율성을 향상시키기 위해 팀 단위로 실습한다.

평가 준거

- 평가자는 학습자가 학습 목표를 성공적으로 달성하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가해야 한다.

학습 내용	학습 목표	성취수준		
		상	중	하
게임 플레이 이벤트 설계 및 구현	- 게임 기획에 따른 게임 플레이 이벤트 설계를 할 수 있다.			
	- 캐릭터의 아이템 인벤토리, 아이템 교체, 아이템 효과 반영에 대한 이벤트를 구현할 수 있다.			
상호작용 이벤트 프로그래밍 구현	- 사용자 캐릭터와 타 사용자 및 비사용자 캐릭터 사이의 상호 작용에 대한 이벤트를 구현할 수 있다.			
	- 사용자 캐릭터와 지형 및 게임 오브젝트 사이의 상호 작용에 대한 이벤트를 구현할 수 있다.			
	- 사용자 캐릭터의 행동에 대한 보상 이벤트를 구현할 수 있다.			

평가 방법

- 포트폴리오

학습 내용	평가 항목	성취수준		
		상	중	하
게임 플레이 이벤트 설계 및 구현	- 게임 기획서에 적합한 게임 플레이 이벤트 설계 능력			
	- 아이템 효과를 반영한 이벤트 시스템 구현 능력			
상호작용 이벤트 프로그래밍 구현	- 보상 이벤트 시스템 구현 능력			
	- 캐릭터와 지형 및 게임 오브젝트 상호 작용 이벤트 시스템 구현 능력			

• 서술형 시험

학습 내용	평가 항목	성취수준		
		상	중	하
게임 플레이 이벤트 설계 및 구현	- 게임 플레이 이벤트 알고리즘에 대한 이해 정도			
	- 게임 이벤트 시스템 알고리즘에 대한 이해 정도			
상호작용 이벤트 프로그래밍 구현	- 게임 보상 이벤트 시스템 알고리즘에 대한 이해 정도			
	- 각 구현 항목별 기초 이론과 구현 원리에 대한 이해 정도			

• 평가자 체크리스트

학습 내용	평가 항목	성취수준		
		상	중	하
게임 플레이 이벤트 설계 및 구현	- 게임 이벤트 시스템을 이해하고 있는지 여부			
	- 게임 보상 이벤트 시스템을 이해하고 있는지 여부			
상호작용 이벤트 프로그래밍 구현	- 게임 플레이 이벤트 알고리즘을 설계하는 능력			
	- 게임 보상 이벤트 알고리즘을 구현하는 능력			

피드백

- 포트폴리오
 - 게임 기획서를 바탕으로 한 게임 서버 설계의 흐름을 충실히 이행하고 있는지 평가한 뒤, 평가 결과가 부족한 학습자에 대해서는 설명을 통해 지도한다.
 - 게임 서버의 설계와 개발된 프로그램 사이의 오류를 평가한 뒤, 평가 결과가 부족한 학습자에 대해서는 설명을 통해 지도한다.
- 서술형 시험
 - 게임 기획서에 기반한 각 항목별 구현 알고리즘을 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 다시 설명해 준다.
 - 상호 작용 이벤트 구조에 따른 알고리즘을 설계의 흐름에 따라 이행하였는지 평가한 뒤, 충실히 이행한 학습자에 대해서는 구현을 할 수 있도록 지도한다.
- 평가자 체크리스트
 - 게임 기획을 바탕으로 만든 서버 프로그램 소스를 체크리스트로 평가한 뒤, 평가 결과가 미흡한 학습자에 대해서는 주요 사항을 표시하여 설명해 준다.



- 교육부(2018). 게임 서버 프로그래밍 (LM0803020532, LM0803020533, LM0803020534). 한국직업능력개발원.
- 김기남 · 김혜영(2019). 「부하 분산기법을 적용한 IOCP 모델의 MMORPG 게임서버 구현에 관한 연구」. 『In Proceedings of KIIT Conference』. pp. 612-615.
- 김인홍 · 김경식(2005). 「MMORPG에서 아이템의 효과적인 제작 및 관리 방안에 대한 고찰」. 『한국게임학회 논문지』 5(1). pp. 61-66.
- 성소윤 · 정내훈(2020). 「io_uring I/O 모델을 통한 MMO 게임 서버의 성능개선」. 『한국게임학회 논문지』 20(6). pp. 31~41.
- 심한뢰 · 방진욱 · 김인호(2022). 「포톤 서버를 사용한 멀티게임 구현(Hide & Escape)」. 『한국컴퓨터 정보학회 학술발표논문집』 30(1). pp. 69-70.
- 윤성우(2003). 『IT CookBook, TCP/IP 소켓 프로그래밍』. 한빛미디어.
- 이석호(2017). 『데이터베이스론』. 정익사.
- 차은상 · 김영식(2019). 「IOCP 게임 서버 및 게임 변수 난독화 프로그램 성능 평가」. 『한국게임학회 논문지』 19(6). 71-81.
- 한진욱 · 김혜영(2018). 「IOCP 기반의 모듈화된 게임서버에 관한 연구」. 『In Proceedings of KIIT Conference』. pp. 565~568.
- Unity 매뉴얼. <https://docs.unity3d.com/kr/530/Manual/>에서 2022. 08. 25. 검색.
- Microsoft WindowsServerEssentials. <https://www.microsoft.com/ko-kr/d/windows-server-essentials/dg7gmgf0dvsz?activetab=pivot%3aoverviewtab>에서 2022. 09. 06. 검색.
- Ubuntu 공식 홈페이지(https://design.ubuntu.com/brand/ubuntu-logo/?_ga=2.249578950.1741125463.1662461641-135178105.1662461641)에서 2022. 09. 06. 검색.
- centos 공식 홈페이지(<https://www.centos.org/>)에서 2022. 09. 06. 검색.

NCS학습모듈 개발이력

발행일	2013년 12월 31일		
세분류명	게임콘텐츠제작(08030205)		
개발기관	한국직업능력개발원		
집필진	김경식(호서대학교)		고병희(예원예술대학교)
	김관웅(전남과학대학교)		김덕수(㈜디지털프로그)
	김대진(울산애니원고등학교)		김영진(청강문화산업대학교)
	김도형(서울디지털고등학교)		김용준(㈜엠게임)
	김인홍(DoStudio)		김인성(서울인덕공업고등학교)
	설종원(동서울대학교)		
	안태홍(전남과학대학교)	검토진	
	이기섭(한국게임과학고등학교)		
	이대웅(상명대학교)		
	이주희(㈜엔씨소프트)		
	임진하(㈜편오락실)		
	최석균(㈜나인버드게임즈)		*표시는 대표집필자임
발행일	2022년 12월 31일		
학습모듈명	게임 네트워크와 클라이언트 프로그래밍(LM0803020534_18v4, LM0803020542_18v1)		
개발기관	상명대학교 천안산학협력단, 한국직업능력연구원		
집필진	유창석(경희대학교)*		이수진(경기게임마이스터고등학교)
	강민정(홍익대학교)		이정엽(순천향대학교)
	김세윤(REVE A.V.C)		장병희(성균관대학교)
	김수완(성균관대학교)		전종우(단국대학교)
	안혜영(㈜딜리셔스게임즈)	검토진	
	오영욱(가천대학교)		
	이현진(㈜딜리셔스게임즈)		
	조수진(엠지미디어디자인)		
	조현재(신구대학교)		
	주진영(편달)		*표시는 대표집필자임

게임 네트워크와 클라이언트 프로그래밍 (LM0803020534_18v4, LM0803020542_18v1)

저작권자	교육부
연구기관	한국직업능력연구원
발행일	2022. 12. 31.
ISBN	979-11-339-9098-6

※ 이 학습모듈은 자격기본법 시행령(제8조 국가직무능력표준의 활용)에 의거하여 개발하였으며, NCS통합포털사이트(<http://www.ncs.go.kr>)에서 다운로드 할 수 있습니다.



www.ncs.go.kr