

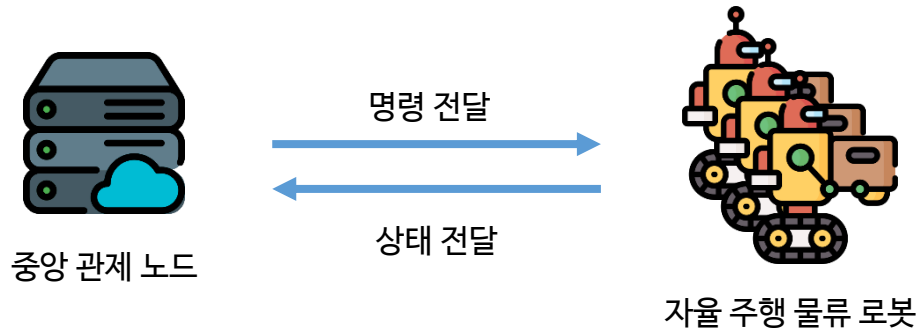
스레드 관리: 자동화 물류 센터 시뮬레이터 개발

Operating System Project#1

자동화 물류 센터 시뮬레이터 개발

◆ 프로젝트 개요

- 중앙 관제 노드와 자율 주행 물류 로봇들을 통해 물류 창고 자동화
- 자율 주행 물류 로봇들이 물건을 적재 후 정확한 위치에 하역하는 시나리오 수행



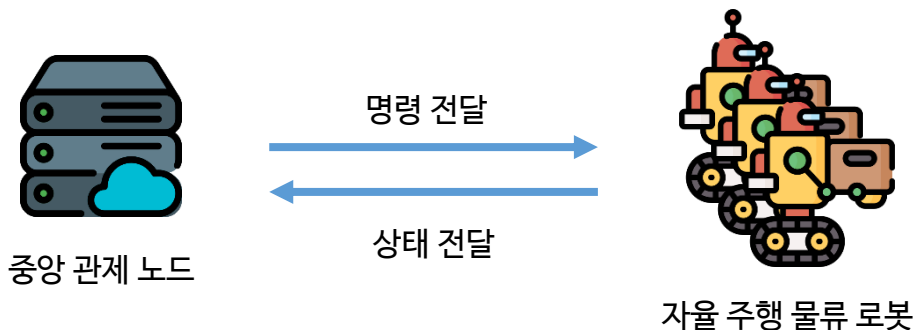
X	X	X	X	X	X	X
A			7			X
X		1	X	4		X
B		2	X	5		X
X		3	X	6		X
C					S	X
X	X	X	X	X	W	X

- 시뮬레이터에서 사용하는 물류 센터는 7x7 크기의 2차원 배열로 구성 (aw_manager.h/.c에 정의됨)
 - 1~7 : 물건 적재 장소
 - A~C : 하역 장소
 - S(Start) : 시작 장소
 - W(Wait) : 대기 장소
 - X : 이동 불가 장소

자동화 물류 센터 시뮬레이터 개발

◆ 프로젝트 개요

- 중앙 관제 노드와 자율 주행 물류 로봇들을 통해 물류 창고 자동화
- 자율 주행 물류 로봇들이 물건을 적재 후 정확한 위치에 하역하는 시나리오 수행



X	X	X	X	X	X	X
A			7			X
X		1	X	4		X
B		2	X	5		X
X		3	X	6		X
C					S	X
X	X	X	X	X	W	X

- 중앙 관제 노드
 - 물류 로봇이 이동할 수 있는 조건을 고려하여 모든 물류 로봇들이 각자 하역 장소 까지 도달하게 해야함
 - 모든 물류 로봇에게 상태 정보를 전달 받을 때까지 대기 후 물류 로봇들의 상태 출력 (요구사항 4)
 - 이후 상태 정보들을 기반으로 각 물류 로봇에게 대기 혹은 상하좌우 이동 행동 지시 후 단위 스텝 증가(요구사항4)
 - 행동 지시 후 모든 물류 로봇들을 Unblock 필수 (요구사항 3)
 - 모든 물류 로봇이 물건을 운송하였으면 시뮬레이션 종료
- 자율 주행 물류 로봇(이하 물류 로봇)
 - 해당 물류 로봇이 요구하는 물건 적재 지역을 거쳐 정해진 하역 장소까지 도달해야 함
 - 초기 물류 로봇은 W에 위치해야 함 (요구사항 1)
 - 모든 물류 로봇은 한 번의 단위 스텝 중에 대기 혹은 이동 중 하나의 행동만 가능 (요구사항 4)
 - 상하좌우 중 한 곳으로만 이동 가능
 - 이동할 위치에 다른 물류 로봇이 있을 경우 이동 불가 (단, 하역 장소, 대기 장소에서는 중복 가능)
 - 적재할 물건이 있는 장소 외 다른 물건이 있는 장소에 이동 불가
 - 물류 로봇은 중앙 관제 노드가 정해진 행동을 취한 후 자신의 상태 정보를 관제 노드에게 전달 후 Block(요구사항 3&4)

자동화 물류 센터 시뮬레이터 개발

◆ 프로젝트 요구사항

- 0. Pintos 환경에서 빌드 및 실행
- 1. 시뮬레이터 실행 파라미터 입력 값을 파싱하여 중앙 관제 노드 및 물류 로봇들에 대한 스레드 생성
- 2. 스레드 간 데이터 전송을 위한 Block, Unblock 함수 구현
- 3. block_thread 및 unblock_threads를 통한 스레드 간 데이터 전달 시스템 구현
- 4. 중앙 관제 노드를 통한 N개의 물류 로봇 제어
- 5. 물류 센터에 동시에 여러 물류 로봇 제어
- 6. 보고서 작성

자동화 물류 센터 시뮬레이터 개발

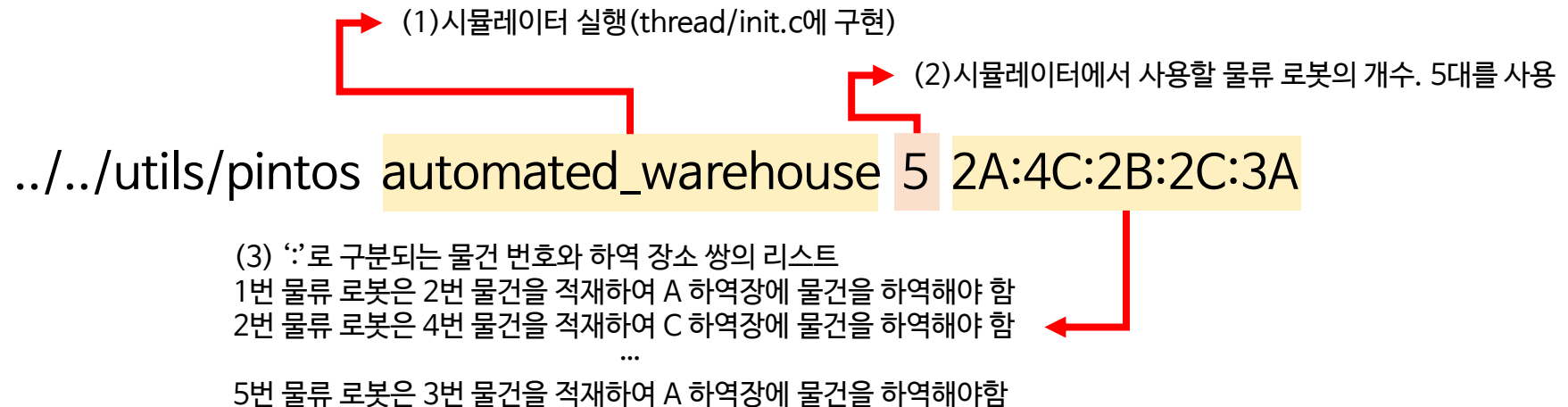
◆ 프로젝트 요구사항

- 0. Pintos 환경에서 빌드 및 실행
 - 본 프로젝트는 Pintos 기반으로 수행되어야 함
 - projects/automated_warehouse 디렉토리의 파일을 수정 및 사용해야 하며 추가적인 파일 생성 가능
 - * automated_warehouse.c 파일의 run_automated_warehouse 함수가 진입점
 - projects/automated_warehouse/aw_manager.h/.c 은 수정 불가
 - 개발 파일 외 다른 파일을 수정할 경우 수정한 근거를 레포트로 작성
 - 다음 파일을 참고하는 것을 권장
 - threads 디렉토리의 init.c, synch.h, synch.c, thread.h, thread.c 및 기타 필요 코드

자동화 물류 센터 시뮬레이터 개발

◆ 프로젝트 요구사항

- 1. 시뮬레이터 실행 파라미터 입력 값을 파싱하여 중앙 관제 노드 및 물류 로봇들에 대한 스레드 생성
 - 시뮬레이터는 threads/build 디렉토리에서 실행하며 (2), (3) 파라미터를 파싱하여 사용할 물류 로봇의 정보를 획득해야 함
 - (1) project/automated_warehouse.c 에 구현된 시뮬레이터 실행을 위한 파라미터
 - (2) 물류 로봇의 개수
 - (3) 각 물류 로봇이 필요로 하는 물건의 번호와 하역 장소를 가지는 리스트
 - ‘:’로 구분되며 예시에서 총 5개의 물건 번호와 하역 장소의 쌍을 가짐
 - 시뮬레이터의 (2) (3) 파라미터는 채점 시에 달라질 수 있음



자동화 물류 센터 시뮬레이터 개발

◆ 프로젝트 요구사항

- 1. 시뮬레이터 실행 파라미터 입력 값을 파싱하여 중앙 관제 노드 및 물류 로봇들에 대한 스레드 생성
 - 중앙 관제 노드와 N대의 자율 주행 물류 로봇들의 스레드를 생성해야 함
 - 중앙 관제 노드
 - 1대의 중앙 관제 노드 생성(메인 스레드로 활용 가능)
 - 물류 로봇
 - 입력 값에 따라 N대 생성
 - i번 물류 로봇의 스레드 이름은 “Ri”으로 명명 ex) 1번 물류 로봇 스레드의 이름은 “R1”
 - 관련 코드는 threads/thread.h, threads/thread.c 파일 참고
 - 스레드 생성 및 block, unblock, yield 등과 같은 상태 관리 함수 존재

자동화 물류 센터 시뮬레이터 개발

◆ 프로젝트 요구사항

- 2. 스레드 간 데이터 전송을 위한 Block, Unblock 함수 구현
 - 요구사항 3의 운송 로봇과 중앙 관제 노드 간의 데이터 전달을 위해 projects/automated_warehouse/aw_thread.h/.c에 정의된 빈 함수 block_thread, unblock_threads 구현해야 함
 - block_thread
 - Block 시킨 스레드를 struct list blocked_threads에 저장해야함
 - 현재 Block만 되는 기능만 구현되어 있으므로 수정해야함
 - unblock_threads
 - struct list blocked_threads에 저장되어 있는 Block된 스레드들을 전부 Unblock
 - 선점을 막기 위해 intr_disable(), intr_set_level(-)가 필요하며, 이 두 개는 반드시 쌍을 맺어야함
 - intr_disable 만 호출하고 intr_set_level을 호출하지 않으면 선점이 불가능한 상황 발생
 - 참고
 - threads/synch.h/.c 에 있는 sema_up, sema_down 함수를 참고

자동화 물류 센터 시뮬레이터 개발

◆ 프로젝트 요구사항

- 3. block_thread 및 unblock_threads를 통한 스레드 간 데이터 전달 시스템 구현
 - 스레드 간 데이터의 일관성을 보장하기 위해 데이터 전달 시스템 외 데이터 공유를 제한해야 함
 - aw_message.h/.c 파일에 작성된 message, message_box 구조체를 활용하여 전달 시스템 구현 필요
 - 개발의 편의를 위해 중앙 관제 노드로부터 오는 메시지, 물류 로봇으로부터 오는 메시지를 구분하지 않고 통합하여 관리
 - message, message_box의 수정이 필요할 경우 해당 방식을 유지하는 것을 권장
 - boxes_from_central_control_node에 각 물류 로봇들이 중앙 관제 노드로부터 받는 메시지 함 배열을 동적 할당해주어야 함
 - boxes_from_robots에 중앙 관제 노드가 각 물류 로봇들에게 받는 메시지 함 배열을 동적 할당해주어야 함
 - 데이터 전달 제어 방식 정의 필요
 - 물류 로봇이 중앙 관제 노드로 데이터를 전달한 후에는 Block되어야 함
 - 중앙 관제 노드가 모든 물류 로봇에게 데이터를 전달 받은 후에는 Block 상태의 물류 로봇 스레드들을 Unblock 해야 함

자동화 물류 센터 시뮬레이터 개발

◆ 프로젝트 요구사항

• 4. 중앙 관제 노드를 통한 N개의 물류 로봇 제어

– 물류 로봇 제어 필요

▪ 요구 사항

- 물류 로봇 배열은 automated_warehouse.c에 “robots” 로 선언되어 있으며 동적 할당 필요
- 해당 물류 로봇이 요구하는 물건 적재 지역을 거쳐 정해진 하역 장소까지 도달해야 함
- 초기 물류 로봇은 W에 위치해야 함
- 모든 물류 로봇은 한 번의 스텝 중에 대기 혹은 이동 중 하나의 행동만 가능해야 함
- 물류 로봇이 하역 장소나 대기 장소에 있어도 상태 정보를 전달 필수

▪ 참고 사항

- 물류 로봇에 대한 구조체는 robot.h/.c에 정의

– 중앙 관제 노드 제어 필요

- 물류 로봇이 이동할 수 있는 조건을 고려하여 모든 물류 로봇들이 각자 하역 장소까지 도달하게 해야함
 - 한번에 하나씩 물류 로봇을 제어해서 미션 수행 가능
- 모든 물류 로봇에게 상태 정보를 전달 받을 때까지 대기 후 aw_manager.h/.c에 정의된 print_map(-) 호출
 - print_map을 수정하거나 호출 도중 Block된 스레드가 없을 시 감점
- 이후 상태 정보들을 기반으로 각 물류 로봇에게 대기 혹은 상하좌우 이동 행동 지시 후 aw_manager.h/.c에 정의된 increase_step() 호출
 - 모든 물류 로봇들을 Unblock 필수
- 모든 물류 로봇이 물건을 운송하였으면 시뮬레이션 종료

자동화 물류 센터 시뮬레이터 개발

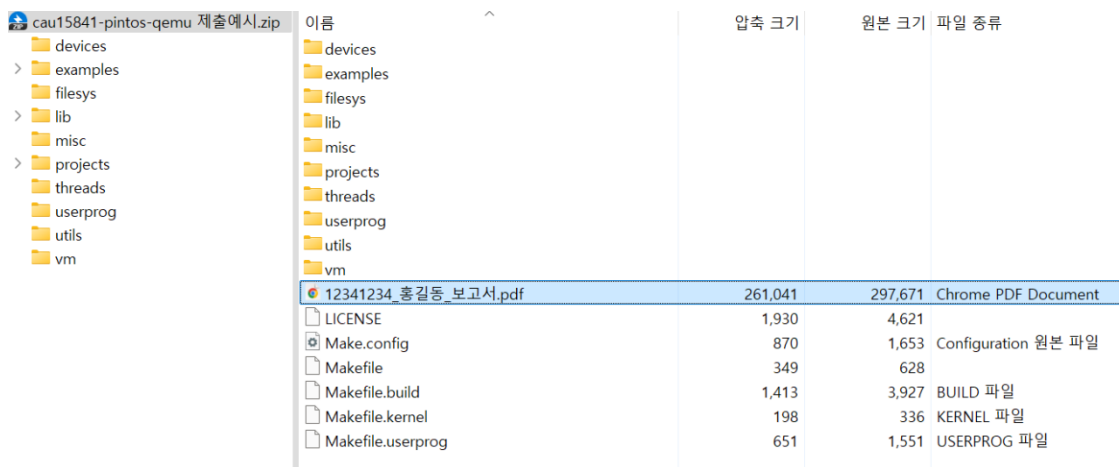
◆ 프로젝트 요구사항

- 5. 물류 센터에 동시에 여러 물류 로봇 제어
 - 물류 센터에 동시에 여러 물류 로봇을 제어할 경우 가산점 부과
 - 동시 제어시에도 물류 로봇의 이동 조건을 어길 시 감점
- 6. 보고서 작성
 - 본인이 작성한 코드에 대한 구현 방법, 트러블 슈팅 과정, 결과에 대해 설명하는 보고서 작성

자동화 물류 센터 시뮬레이터 개발

◆ 제출 시 주의사항

- `projects/automated_warehouse/aw_manager.h/.c` 은 수정 불가
- make clean 후 제출 필요
- 다음과 같이 보고서와 함께 압축 후 제출



The screenshot shows a file explorer window with a directory tree on the left and a list of files on the right. The directory tree includes folders like devices, examples, filesys, lib, misc, projects, threads, userprog, utils, and vm. The file list on the right contains the following data:

이름	압축 크기	원본 크기	파일 종류
12341234_홍길동_보고서.pdf	261,041	297,671	Chrome PDF Document
LICENSE	1,930	4,621	
Make.config	870	1,653	Configuration 원본 파일
Makefile	349	628	
Makefile.build	1,413	3,927	BUILD 파일
Makefile.kernel	198	336	KERNEL 파일
Makefile.userprog	651	1,551	USERPROG 파일

Pintos

◆ Pintos

- 커널 학습을 위한 교육용 운영체제

◆ Pintos 빌드 및 실행

- 1. Ubuntu 18.04 LTS 이미지를 다운받아 VMWare Workstation Player의 가상 노드를 만들어 띄움
- 2. 다음 명령어를 사용하여 GCC, GNU Binutils, Perl, GNU make, 그리고 QEMU 패키지를 설치
 - `sudo apt install build-essential gcc binutils perl qemu gdb`
- 3. 프로젝트 소스를 다운받아 원하는 위치에 압축을 푼 후 프로젝트 디렉토리로 이동 후 빌드를 수행
 - `cd threads`
 - `make`
- 4. 빌드가 완료 되면 다음 명령어를 통해 테스트 프로그램을 실행
 - `cd build`
 - `../utils/pintos automated_warehouse 5 2A:4C:2B:2C:3A`

자동화 물류 센터 시뮬레이터 개발

◆ 과제 관련 문의

- 조민규 조교 (mgjo@cslab.cau.ac.kr)
- 연구실 위치 : 208-530