

팀장 : 이 규 봉

조원(개인, 팀)

학번	이름	맡은 역할	비고
201502085	이규봉	개인 프로젝트 진행	

Github Repo 주소 - <https://github.com/jopemachine/Compiler-HW>

주제

: mini C언어 => x86 어셈블리어로의 변환

주제 선정 이유

: HW 05-2를 자바 바이트 코드로 진행했었는데, 이 과정에서 c언어가 어셈블리어로 어떤 식으로 변환되어야 하는지 생각해 볼 수 있어 재미있는 경험이 되었습니다. 다만 제가 평소에 흥미가 있었던 어셈블리어는 자바 바이트 코드가 아니라 x86 MASM 이었기 때문에 x86 MASM으로 작성해 보고 싶어 주제를 선정했습니다.

주제 적용 시 향상되는 점 및 예상 결과

: 주어진 개발기간이 짧아 실제로 쓸만한 컴파일러를 만들기엔 역부족일 것으로 예상됩니다. 하지만 평소에 어셈블리어에 관심이 있었기 때문에, 텀프로젝트 종료 이후에도, 개발을 진행해보려 계획 중이며, 기초적인 코드 작성에 신경을 기울일 계획입니다.

구현 일정 계획 (일자로 마일 스톤 표시)

: 실제로 개발을 시작하기 전엔 구체적인 내용을 알기 어려워 간단히만 표시하였습니다.

작업내용	11/27	11/30	12/01	12/04	12/05	12/07
X86 어셈블리어로 변환 (기초)						
제어문, 함수 구현						
버그 수정						

- 팀장 연락처 (이메일)

- 이규봉 : gravityc95@gmail.com

hw 05-3

- 자유로운 주제를 가지고 hw05-2 의 결과를 컴파일러를 수정/개선해본다.

mini C 와 JAVA byte code 둘 중 하나는 반드시 활용한다.

예)

- mini C를 x86이나 x64, ARM 등의 어셈블리로 변환하는 컴파일러 만들기
- mini C를 GCC GIMPLE이나 LLVM IR 등 JAVA byte code 가 아닌 다른 중간 코드로 변환하는 컴파일러 만들기
- mini C나 JAVA byte code의 제어 흐름 그래프를 만들어 화면에 보여주기
- mini C에서 함수 호출 그래프를 만들어 화면에 보여주기
- 변환된 JAVA byte code에 최적화를 해주는 컴파일러 만들기
- 타입 검사가 추가된 mini C to JAVA byte code 컴파일러 만들기

기타 수업 중 배운 기법들을 활용해서 miniC나 JAVA byte code 조작에 관련 있는 과제,

또는 miniC to JAVA byte code 변환기를 개선하는 과제

(애매한 경우 메일로 사전 협의 요망-eschough@cnu.ac.kr)

- 팀구성

- 원칙적으로 2-3인으로 구성된 팀별 과제이다.
- 원하면 단독으로 할 수도 있다.
- 팀원 모두가 hw05-02 를 제출한 상태여야 팀구성이 가능하다.
- 팀을 구성하는 경우와 개인 단독 수행의 점수상 차이는 없다.
- 팀으로 구성한 제출물에 대해서는, 역할에 대한 면밀한 조사를 하여 팀원간 역할 배분이 상식 이하로 불공평할 때는 경우에는 크게 (경우에 따라 hw5-3의 배점 이하로) 감점한다.
즉, 현실적으로 두 사람 이상이 함께 시너지를 내어서 잘 할 수 있는 경우에만 전략적으로 팀으로 하여 각자의 로드를 줄이고, 그렇지 않는 경우는 권하지 않는다.
- 나머지 팀원 모두가 동의할 시 팀원을 방출할 수 있다.
- 팀으로 시작하더라도 진행하다가 개인으로 변경할 수 있다..

기타

- Item1 ~ Item 3 모두, 윤리적인 문제가 발생했을 시에는 0점 처리 또는 감점이 가능함
- Item2, Item 3는 유사도 검사 예정 (역대 제출물들과도 비교함)