

project plan

Debugging Narae



**-변 경 이 력-**

|  |  |  |  |
| --- | --- | --- | --- |
| 일자 | 버전 | 변경 내역 | 작성 |
| 2017-09-26 | v0.01 | 초안 작성 | 조은재 |
| 2017-10-07 | v0.02 | 문서 양식 수정 | 김정원 |
| 2017-10-07 | v0.02.1 | 문서 수정 | 조은재 |
| 2017-10-14 | v0.03 | 문서 수정 | 임다빈 |
| 2017-10-14 | v0.04 | 문서 수정 | 조은재 |
| 2017-10-15 | v0.05 | 스프레드시트 수정 | 조은재 |
| 2017-11-11 | V1.00 | Google word 에서 MS office word로 옮김.  문서 버전 관리 이슈의 전환점을 고려하여  v1.00으로 업데이트 | 김정원 |
|  | V1.01 |  |  |

목차

**1. 개요1**

1.1 목적1

1.2 주요 일정1

1.3 조직2

1.3.1 조직도2

1.3.2 역할 및 책임2

1.4 도구3

**2. 일정4**

2.1 DNarae 계획 스프레드시트5

**3. 사용자 인터페이스 초안6**

3.1 웹6

3.1.1 웹 페이지 인덱스 화면6

3.1.2 디버거 화면7

3.2 웹 프레임워크9

3.2.1 Python Flask9

3.3 VPS9

**4. 디버거10**

4.1 기능10

4.2 Disassembler10

4.3 Hex Dump + Editor11

4.4 Breakpoint?

4.5 Register View?

1. 개요
   1. 목적

본 프로젝트의 목적은 인터넷 웹 브라우저 환경에서 바로 사용 가능한 디버거 프로그램을 제작하는 것이다.

여기서 디버거(debugger)는 메모리에서 실행 중인 프로그램 코드에 중단점을 지정하여 프로그램 실행을 정지하고 메모리에 저장된 값을 살펴 보며, 해당 프로그램의 소스 코드를 디스어셈블하여 보여준다. 또한 디버깅 기능을 위해 실행을 재개하거나, 코드를 단계적으로 실행하는 등의 동작을 한다.

이 프로젝트는 VPS 환경에서 파이썬 웹 프레임워크인 flask를 이용해 파이썬으로 만든 디버거 서비스를 웹 브라우저에서 실행하는 것이며 디버거의 주요 기능으로 기초적인 CLI 디스어셈블러를 만드는 것을 목표로 한다.

가장 이상적인 목표는 GUI 디버거 애플리케이션을 웹에서 실행할 수 있게 만드는 것이다.

* 1. 주요 일정

|  |  |  |
| --- | --- | --- |
| 단계 | 일정 | 산출물 |
| 설계 | 2017-10-11 | 프로젝트 계획서.doc |
| 중간보고 | 2017-10-25 | 프로젝트 계획서.doc  설계서.doc |
| 구현 | 2017-10-31 | 전용 웹 페이지  프로그램 소스 코드 |
| 평가 | 2017-11-25 | 테스트 케이스.doc  & 테스트 결과 보고서.doc |
| 최종보고 | 2017-11-30 | 최종 발표.ppt |

* 1. 조직
     1. 조직도

Debugger Designer

이지연

Project Manager

조은재

web Designer

이나윤

Developer

임다빈

Tester

김정원



* + 1. 역할 및 책임

|  |  |  |
| --- | --- | --- |
| 이름 | 역할 | 책임 |
| 조은재 | Project Manager  Code Developer | 개발 계획 / 일정을 총괄하여 전반적인 진행 결과를 모니터링하고 컨트롤하며 구성원들이 능력을 잘 발휘 할 수 있도록 일감을 배분하고 준비시킨다.  디버거 제작을 맡는다. |
| 이지연 | Debugger page Designer | 사용자에게 보여줄 디버거 페이지에 대한 레이아웃을 고안하고 완성한다.  4개의 프레임에 디버거 소스 파일들을 업로드하는 기능을 추가한다. |
| 이나윤 | Web page Designer | 사용자에게 보여줄  웹페이지에 대한 기본 레이아웃을 고안하고 완성한다. |
| 임다빈 | Web server & Web framework Developer | 기술 고문 및 VPS를 통해 환경을 구축하고 운용, 디버거와 웹 서버와의 데이터 통신 시스템을 완성한다. |
| 김정원 | Tester & Document manager | 프로젝트의 문서에 대해 전반적으로 검토하고 수정하며 프로젝트 결과물에 대해 테스트한다.  테스트 결과에 대해 모든 데이터를 보고서로 산출한다. |

* 1. 도구

|  |  |  |
| --- | --- | --- |
| 용도 | 이름 | 설명 |
| 형상 관리 | Github | 모든 문서의 버전 관리와 작업자들 간의 공유에 활용한다. Github를 통해 소스코드를 관리한다. |
| 의사소통 | Discode | 음성 채팅 및 PC 화면 공유로 온라인 회의나 논의 및 정보 공유 활용 |
| 코드 작성 | Python 3.6.3  flask  Html / Css | 소스코드 작성 간에 프로그래밍 언어와 도구의 표준을 정한다. |
| 문서 도구 | MS Office Word  MS Office Excel  MS Office PowerPoint | 모든 문서 도구와 양식을 통일 한다.  각종 다이어그램과 차트는 Google Presentation와 Google Spreadsheet를 활용한다. |
| 제출 도구 | Corn | Github과 구글독스를 통해 버전 관리 된 보고서를 취합하여 요약 후 업로드한다. |

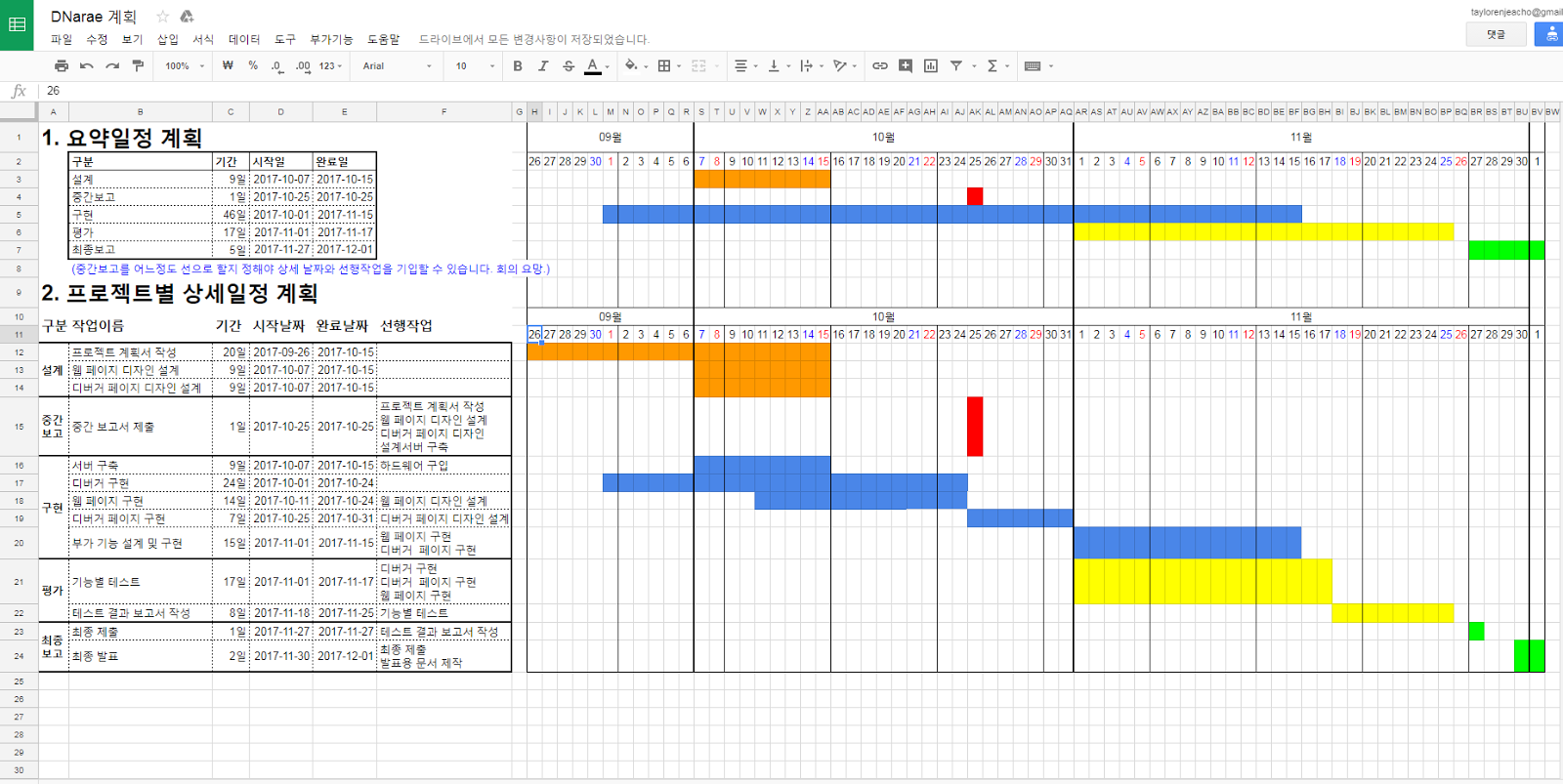
|  |  |
| --- | --- |
| 날짜 | 내용 |
| 09/25 | 오프라인 첫 회의 ; 오프라인 모두 참석 / 프로젝트의 정확한 목표 설정. |
| 09/26 | OT |
|  | 온라인으로 대략적인 계획 문서 배부  -> 각각 하는 일에 대한 디테일 각자 추가 |
| 09/26~10/12 | 각자 개발환경 구축 |
| 추석 이후 | 오프라인 회의 ; 개발환경 점검 / 앞으로의 계획에 대한 회의 |
|  | 중간 점검 확인 & 지원금 받기 |
| 10/13 | 오프라인 회의 ; 기본 환경 구축, VPS 선택, Github 저장소 추가 및 사용법 설명. |
| 중간고사 이후 | 오프라인 회의 ; 서버 구매 / 홈페이지 제작 시작 / 디버거 코드 이해 |
|  | 오프라인 회의 ; 디테일에는 신경쓰지 말고 홈페이지와 디버거 모두 실행 가능하게만. |
|  | 코드와 디자인과 설계 등 수정. |
|  | 디버깅..... |
|  | //(이 기존 자료 참고로 밑에 일정 수정 하시고 불필요하시면 그 때 삭제) |

1. 일정

|  |  |  |
| --- | --- | --- |
| 작업 명 | 소요시간 | 선행작업 |
| 프로젝트 계획서 작성 | 16일 (09/26 ~ 10/15) | - |
| 웹 페이지 디자인 설계 | 05일 (10/07 ~ 10/15) | - |
| 디버거 페이지 디자인 설계 | 05일 (10/07 ~ 10/15) | - |
| 서버 구축을 위한 사전 조사 | 05일 (10/07 ~ 10/15) | VPS의 개념과 사용법 논의 |
| 기본 디버거 구현 | 24일 (10/01 ~ 10/24) | 개발환경 구축 |
| 중간 보고서 제출 | 10/25 | 프로젝트 계획서 작성  웹 페이지 디자인 설계  디버거 페이지 디자인 설계  VPS 구축 |
| 웹 페이지 구현 | 14일 (10/11 ~ 10/24) | 웹 페이지 디자인 설계 |
| 디버거  페이지 구현 | 07일 (10/25 ~ 10/31) | 디버거 페이지 디자인 설계  디버거 구현 |
| 부가 기능 설계 및 구현 | 15일 (11/01 ~ 11/15) | 웹 페이지 구현  디버거 렌더링 페이지 구현 |
| 기능별 테스트 | 17일 (11/01 ~ 11/17) | 디버거 구현  웹 페이지 구현  디버거 렌더링 페이지 구현 |
| 테스트 결과 보고서 작성 | 08일 (11/18 ~ 11/25) | 기능별 테스트 |
| 최종 제출 | 01일 (11/27 ~ 11/27) | 테스트 결과 보고서 작성 |
| 최종 발표 | 02일 (11/30 ~ 12/01) | 최종 보고 및 제출  발표용 문서 제작 |

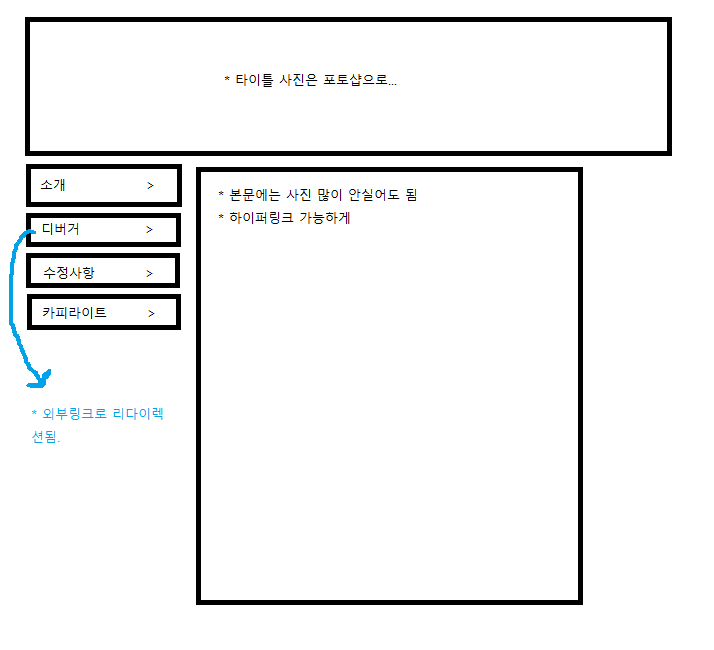
* 1. DNarae 계획 스프레드시트

(google 스프레드시트에 원본 있습니다.)

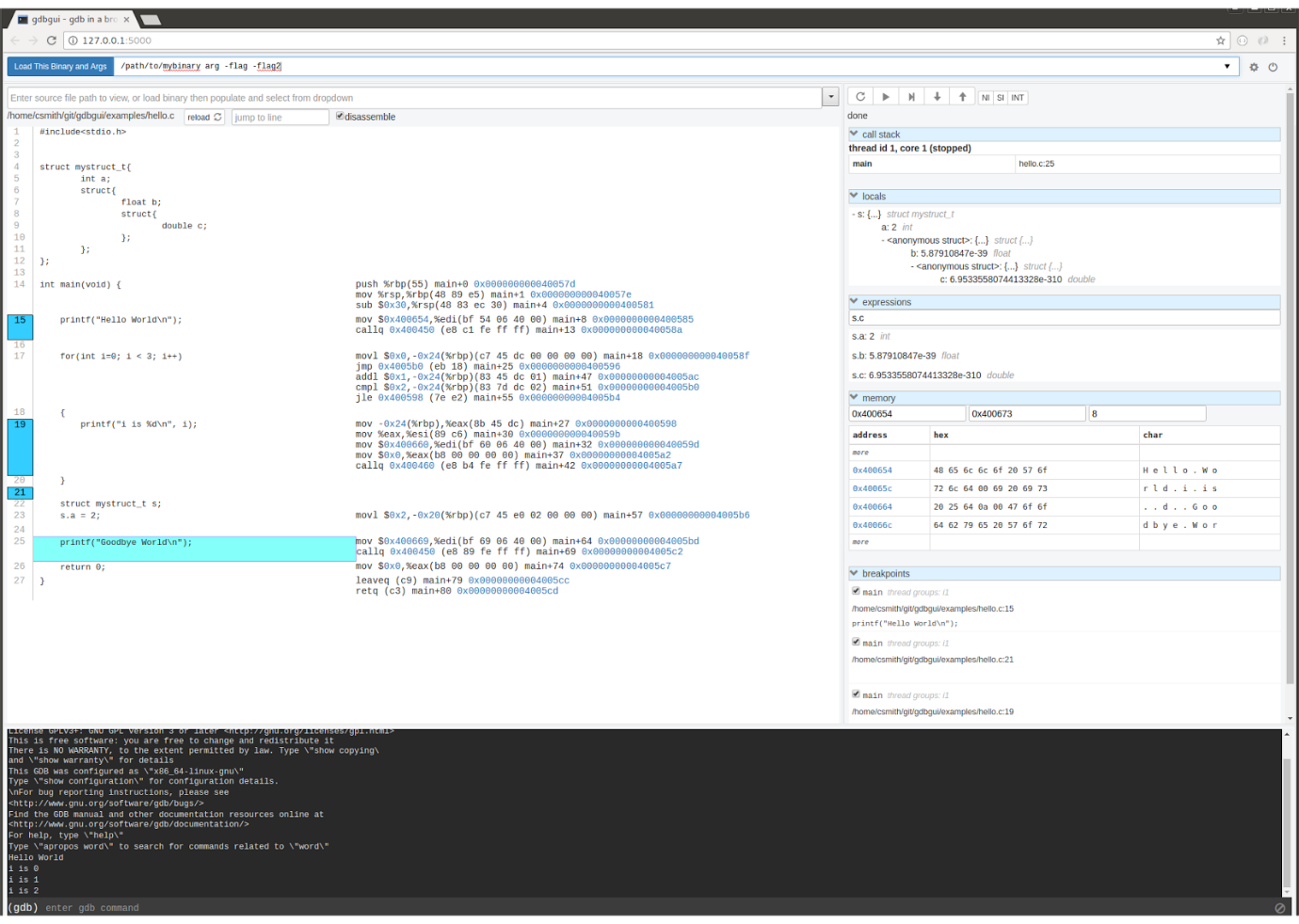


링크 : <https://docs.google.com/spreadsheets/d/1ECTS9jNV9uSauGdVGnLbwEgflHnzlIvZT5CmzYrvoQg/edit?usp=sharing>

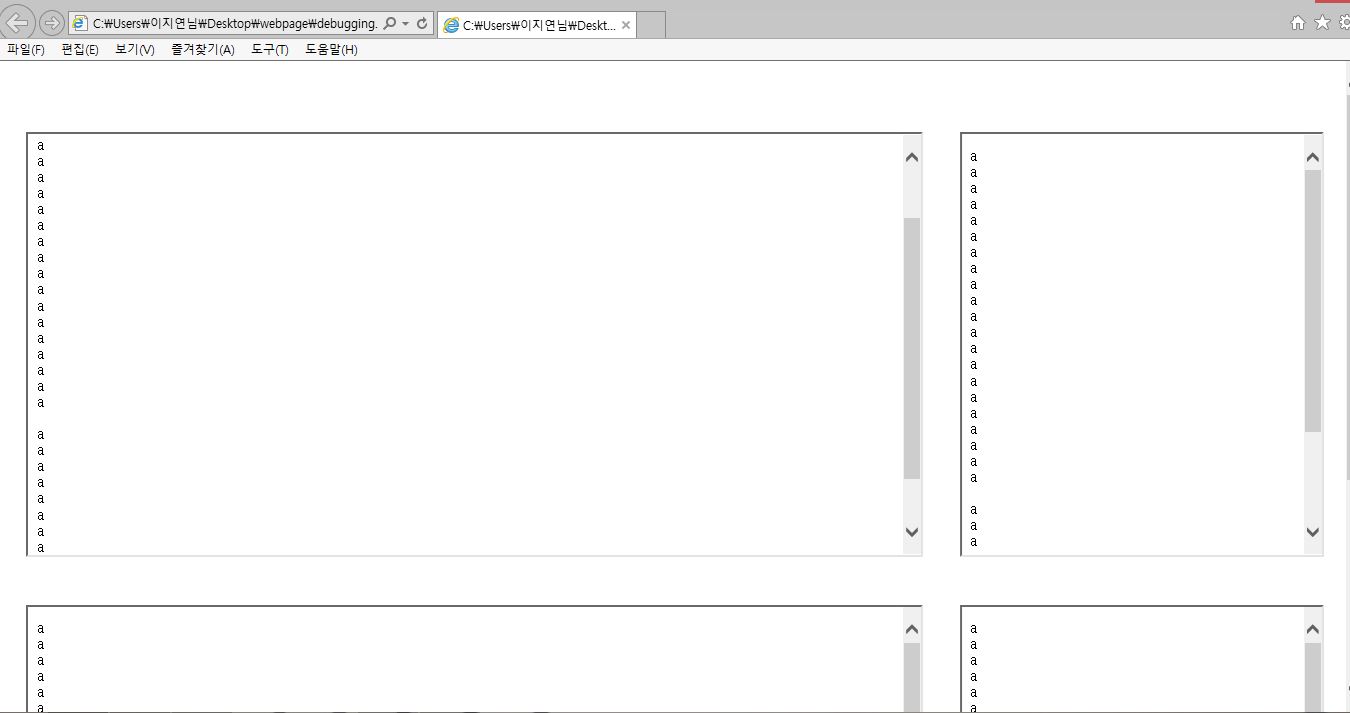
1. 사용자 인터페이스 초안
   1. 웹
      1. 웹 페이지 인덱스 화면



* + 1. 디버거 화면



디버거 페이지 레이아웃



EditPlus 문서 편집기에서 html을 사용하여 디버거 페이지를 작성하고 css로 디자인한다.

각 프레임에 디버거 소스 파일을 업로드한다.

<!DOCTYPE HTML>

<HTML>

<HEAD>

     <style type='text/css'>

         .a{width:67%; height:400px; overflow:scroll;}

         .b{width:27%; height:400px; overflow:scroll;}

 </style>

</HEAD>

<BODY>

<br><br><br>&nbsp;&nbsp;&nbsp;

   <iframe class='a' name="mypage" src="a.txt"></iframe>

&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<iframe class='b' name="mypage" src="a.txt"></iframe><br><br><br>

&nbsp;&nbsp;&nbsp;

<iframe class='a' name="mypage" src="a.txt"></iframe>

&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<iframe class='b' name="mypage" src="a.txt"></iframe>

</BODY>

</HTML>

* 1. 웹 프레임워크
     1. Python Flask

flask란 파이썬으로 구현된 소스코드를 웹서비스로도 사용할 수 있게 해주는 프레임워크이다.

이 프레임워크를 이용해 디버거를 웹브라우저를 통해 렌더링하여 실행 가능하게 한다.

<http://flask.pocoo.org/>

* 1. VPS

VPS란 기존의 웹 호스팅과 다르게 가상 서버를 임대받는 구조로, 물리적으로 현존하는 서버가 아닌 가상으로 제작 된 서버를 제공받는다. 가상이지만 실제 서버와 똑같이 원격 접속 및 관리가 가능하다.

이 프로젝트는 VPS를 통해 네트워크만 연결되어 있다면 사용자가 언제든지 접근 가능한 환경을 만드는 것을 목표로 하며, 코노하(conoha)에서 임대받은 VPS로 환경을 구축한다.

코노하를 선택한 이유?

1. 상대적으로 가격이 저렴하다.
2. VPS 서버 위치를 도쿄로 고를 수 있어서 네트워크 접속 속도가 빠르다.
3. 트래픽이 무제한이다.

https://www.conoha.jp/ko/

1. 디버거
   1. 기능

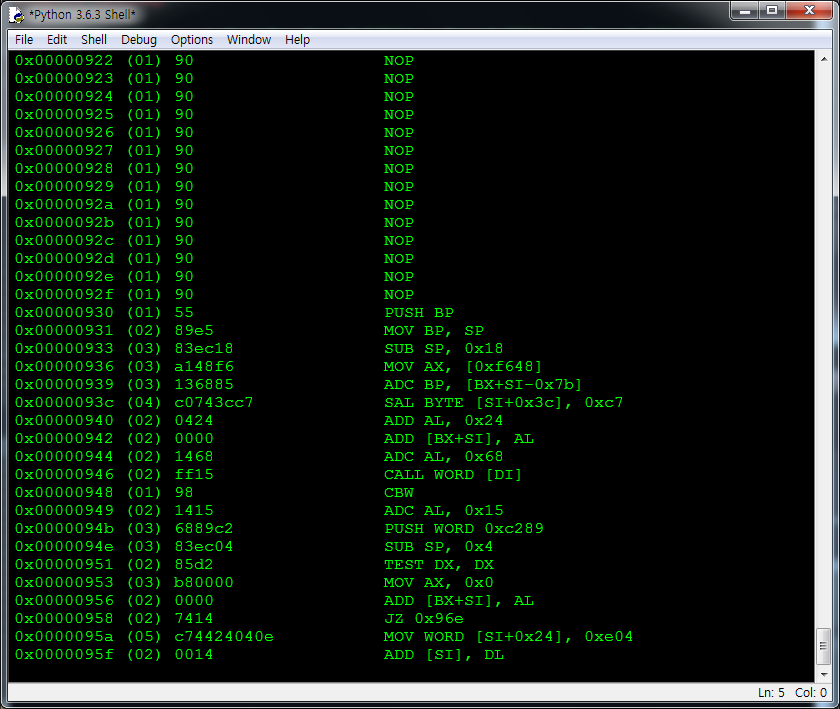
개발은 파이썬으로 한다. 버전은 3.6.3.

CLI 환경으로, Command Line Argument를 이용해서 입출력을 한다.

제공되는 기능은 다음과 같다 :

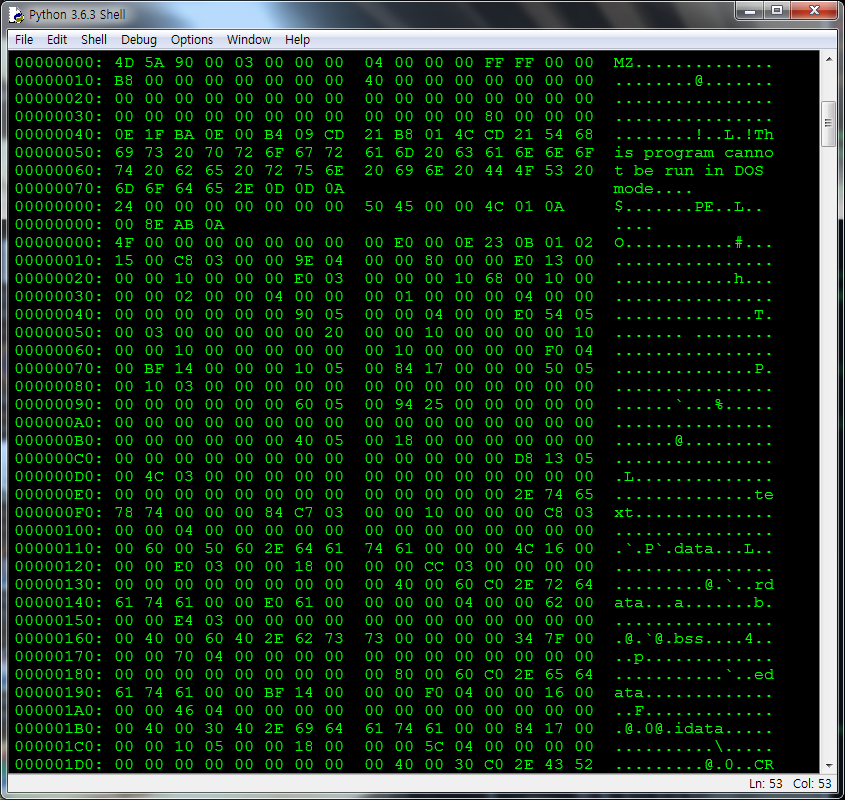
* + 1. Disassembler

기계어를 어셈블리어로 변환해주는 기능을 한다.



* + 1. Hex Dump + Editor

파일에 있는 컴퓨터 데이터를 16진수로 보여주는 기능을 한다. 추가로 편집 기능까지 구현할 예정이다.



* + 1. Breakpoint

실행중인 소프트웨어 프로그램을 조사 또는 편집의 목적으로 의도적으로 잠시 멈추게하는 기능을 한다.

* + 1. Register view

스택과 레지스터에 있는 값을 보여주는 기능을 한다.