

# 어셈블리어언어및실습

## 숙제1

Kon-Woo Kwon  
Department of Computer Engineering  
Hongik University

# 숙제1

---

- Solaris 서버에서 어셈블리 프로그래밍을 하는 방법을 익히기 위한 숙제
  - ❖ 수업시간에 본 예제를 Solaris 환경에서 직접 입력,
  - ❖ gcc를 이용한 어셈블,
  - ❖ gdb를 이용한 디버깅을 실습해 보고 제출하는 것이 숙제의 내용임
- 숙제 제출기한: 9월 29일 화요일 23시 59분까지

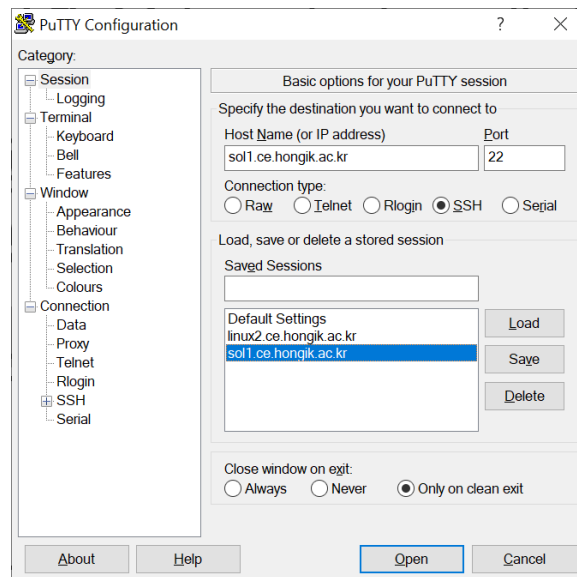
# Solaris 접속방법

## ■ ssh 접속 주소

- ❖ sol1.ce.hongik.ac.kr
- ❖ 계정(ID) / Password 등은 학기초에 일괄 생성되었음
  - ID: 학번,
  - Password: 학번!@
  - 예) 학번이 B123456 이면 ID: B123456 / Password: B123456!@

## ■ ssh 프로그램은 원하는대로 사용 가능

- ❖ 예) Putty



# 숙제 내용

- 다음 코드를 작성

```
        L = 10
        .global main

main:    save %sp, -96, %sp
        mov    %g0, %o0        ! s=0
        mov    1, %l0          ! i = 1
loop:    subcc  %l0, L, %g0      ! if i > 10
        bg     next_r          ! 루프 이탈
        nop
        add    %o0, %l0, %o0    ! s += i
        inc    %l0              ! i++
        ba     loop            ! 후방 분기
        nop
→ next_r: ret
        restore
```

- gdb에서 화살표 위치에 breakpoint를 만들고 %o0 레지스터의 결과값 확인

# 숙제 방법

---

- ssh로 로그인 하면 기본 홈 디렉토리로 오게 됨
- 각 숙제별로 디렉토리를 구분하기 위해 디렉토리 생성
  - ❖ `mkdir hw1`
  - ❖ `cd hw1`
- Vi, Vim, Emacs 등 자신이 원하는 에디터를 사용하여 프로그램 작성
  - ❖ `vim hw1.s`
- 컴파일
  - ❖ `gcc -g hw1.s -o hw1`
- gdb 실행
  - ❖ `gdb hw1`

# **gdb를 사용한 디버깅 실습**

---

1. next\_r 레이블에 breakpoint 설정
  2. next\_r 레이블에 도달할 때까지 프로그램 실행
  3. %o0 레지스터의 값 출력
- 
- 위의 과정을 하는 동안 화면에 출력된 것을 텍스트 파일로 저장하여 소스파일과 함께 제출

# 화면 출력을 텍스트 파일로 저장하기

---

- gdb 실행 시 다음과 같이 실행하면 gdb종료 할 때까지 화면에 출력된 내용이 파일에 저장됨

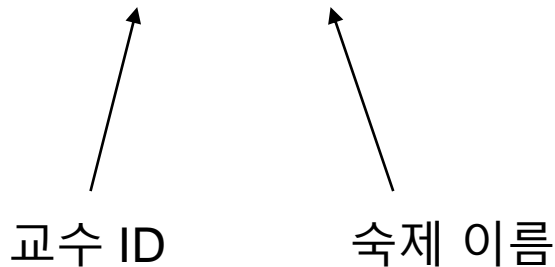
```
gdb hw1 2>&1 | tee output.txt
```

- 화면 출력 파일은 gdb실행 여부를 확인하기 위한 것이므로 완벽하게 정답과 똑같은 출력 형식을 지켜서 제출할 필요는 없음.

# 숙제 제출방법

---

- submit 프로그램 사용하여 제출
  - ❖ 이번숙제에서는 **hw1.s 소스파일**과 **output.txt 파일**을 제출
- 클래스넷을 통해 제출하지 말 것.
- submit 사용방법
  - ❖ 디렉토리에 제출할 파일만 남기고 다른 파일은 다 삭제하거나 다른곳으로 이동시킴
  - ❖ **submit konwoo asm\_01**





# Submit 실행 예

```
B123456@sol1:~/hw1$ ls
hw1.s
output.txt
B123456@sol1:~/hw1$ submit konwoo asm_01
=====
== SUBMIT ver 2.0.3 =====
=====
> Connected at 09/16/20 - 04:47PM.
Normal/Submission is possible.
> Here are descriptions.
* DUE DATE      : 09/29/20 - 11:59PM  <+ 319 H/ 37 M>
* RECEIVER      : konwoo  </lab/KK/konwoo/asm_01/B123456>
* MULTIPLE      : yes
* SUB FOLDER    : no
* FILES IN (/lab/KK/B123456/hw1)
    output.txt hw1.s
* WARNING       : No warning.
> Are you sure to submit? (yes/no) : yes
-----
Total 2 files were transferred.
/lab/KK/konwoo/asm_01/B123456:
total 2
-rwx-----  1 konwoo      KK           377  9월 16일  16:47 hw1.s
-rwx-----  1 konwoo      KK           975  9월 16일  16:47 output.txt
B123456@sol1:~/hw1$
```

# Submit 결과 확인 명령어

```
B123456@sol1:~/hw_test/hw1$ submit konwoo asm_01 -1
=====
== SUBMIT ver 2.0.3 =====
=====
> Connected at 09/16/20 - 05:07PM.
Normal/Submission is possible.
/lab/KK/konwoo/asm_01/B123456:
total 2
-rwx----- 1 konwoo KK 377 9월 16일 16:47 hw1.s
-rwx----- 1 konwoo KK 975 9월 16일 16:47 output.txt
```

# 제출 기한 엄수

---

- 9월 29일 화요일 23시 59분까지 제출
  - ❖ `submit konwoo asm_01`
- 기한 내 제출못했을 경우, 다음과 같이 제출가능하나 감점
  - ❖ `submit konwoo asm_01_late`
    - 9월 30일 수요일 23시 59분까지 제출: (100점 기준) 30점 감점
    - 10월 1일 목요일 23시 59분까지 제출: (100점 기준) 60점 감점
    - 10월 2일 금요일 23시 59분까지 제출: (100점 기준) 90점 감점
    - 이외의 경우 : (100점 기준) 100점 감점