<PA1 REPORT>

전략:

먼저 1. assembly를 2진수 32자리 char문자열로 저장 ->

2. Strtol()로 4자리씩 끊어 16진수 8자리 문자열로 저장 ->

3. 16진수 8자리 문자열을 strtol()로 unsigned int변수에 저장한 후 리턴.

1. 파싱된 tokens이 주어지면 첫자리 토큰인 tokens[0]으로 타입을 3가지로 구분하기로 했습니다.

타입별로 format이 다르기때문에 r과 i를 나눴고 r도 shift연산은 format이 달라 s-type으로 구분했습니다.

(r-type: add, sub, and, or, nor / s-type: sll, srl, sra / i-type: addi, andi, ori, lw, sw, beq, bne)

r-type: opcode, funct, shamt는 미리 값을 저장해놓고 해당하는 값(이진수)을 할당했습니다. Register는 find\_rx함수를 만들어 해당하는 이진수값을 찾아줬습니다.(문자와 숫자가 같이있는 v,a,t,s,k는 t\_dump함수로 따로 처리했습니다.)

find\_rx() : 우선 레지스터이름 뒤에 숫자가 있는경우와 없는경우로 구분했습니다.

zero, at , gp, sp, fp, ra레지스터는 해당하는 이진수 5자리를 바로 반환합니다.

나머지 경우는 t\_dump()라는 함수를 사용한 후 레지스터에 해당하는 10진수를 받아 2진수 문자열로 반환했습니다.

t\_dump() : 레지스터 이름 두번째 자리의 문자(‘0’~’9’)에 따라 번호가 달라지기 때문에 레지스터이름의 앞글자로 케이스를 나누고 각 케이스에 따라 정해진 값 + 두번째 문자(‘0’~’9’)만큼을 10진수로 리턴합니다.

s-type: opcode, funct,register는 r-type과 동일하게 처리했습니다. 입력에서 Shamt에 주어지는 값은 10진수와 16진수 2가지 이므로 입력에서 x가 있는지 찾은 후(x가 있으면 16진수) strtol()로 해당 진수로 읽었습니다. Int sh에 읽은 값을 저장하고 sh를 이진수로 변환해 shamt에 문자열로 저장했습니다.

i-type: opcode, register는 r-type과 동일하게 처리했고 constant의 위치가 sw, lw와 나머지가 다르기때문에 이를 구분해 처리했습니다. Shamt는 s-type과 유사하게 처리했습니다. 차이점은 음수값이 오면 읽은 10진수를 2진수로 변환할때 0과 1을 바꿔 저장한 후 1을더해 two’s complement로 표현했습니다.

1. 1번과정으로 저장된 32자리 이진수를 4개씩 끊어 문자열에 저장한 후 이를 strtol()로 2진수로 읽어to\_x()함수에 넣으면 16진수 한자리 문자를 리턴합니다. 리턴한 값을 차례로 저장해 16진수 8자리 문자열을 만들었습니다.
2. 만들어진 16진수 8자리문자열을 strtol()로 unsigned int 변수에 저장한 후 translate()의 리턴값으로 사용했습니다.

새로 알게 된 내용

1. Strtol()

#include <strlib.h>

long int strtol(const char \*nptr, char \*\*endptr, int base)

strtol()을 사용하면 문자열에 저장된 수를 원하는 진수로 읽어 long타입으로 저장할 수 있다는 것을 알게 되었습니다. Endptr로 문자를 어디까지 읽을 것인가 결정할 수 있고 종료포인터를 설정하고 싶지 않으면 NULL을 사용하면 된다는 걸 알게 되었습니다.

1. 지역변수를 초기화하지 않았을때, OS별 변수초기화가 다를 수 있음

처음 맥북에서 gcc로 컴파일하고 실행했을때 정상적으로 출력이 나와 제출했는데 PASS에서는 모든 값이 이상하게 출력되었습니다. 그래서 window에서 컴파일해 exe파일을 만들고 실행해보니 PASS와 동일하게 오답이 나왔습니다. 코드를 확인해보니 변수중에 초기화하지 않은 변수가 많다는걸 확인했고 모두 적절한 값으로 초기화하고 다시 컴파일했더니 정상적으로 출력이 나왔습니다.

함수가 재호출 되었을때, 새로운 공간에 지역변수가 할당되어 초기화를 따로 하지 않아도 되겠다 생각했었는데, char x\_word[10]을 초기화하지 않고 출력된 결과를 보니 정확히 2칸씩 출력이 밀려있는걸 확인했습니다. 8만큼 사용하고 남은 공간에 새로운 변수를 할당해서 이런 문제가 발생한 것 같습니다.

변수 초기화를 잊지 않도록 하겠습니다.