

CSE3030 어셈블리 프로그래밍 숙제 #3 카이사르(시저) 암호(Caesar cipher)

카이사르 암호

율리우스 카이사르가 사용했다고 하는 암호화 방법을 예를 들어 설명하면 다음과 같다. 사용하는 문자 집합 $S = \{a,b,c,d,e,f\}$ 라고 하고, 이들 문자들에 순서를 준 문자열 리스트를 $L = [a,b,c,d,e,f]$ 라고 하자(문자들의 순서는 임의로 정할 수 있으나 일단 정해지면 고정된다). 여기에 key 값으로 양의 정수 k 가 주어진다.

이제 S 의 문자로 구성된 문자열 $P = [x_0, x_1, \dots, x_{(n-1)}]$ 이라면, x_i 에 대해 cipher 문자 y_i 는 다음과 같이 구한다:

리스트 L 에서 x_i 가 index h 인 위치에 있다고 하면, y_i 는 L 에서의 index가 $h+k$ 인 문자로 한다. $|L| = n$ 이라고 할 때, 만일 $h+k \geq n$ 이면 cyclic하게 L 의 처음부터 $h+k-(n-1)$ 번째 문자로 정한다(즉, L 에서의 index는 $h+k-n$ 이다).

예를 들어 위에 보인 $L = [a,b,c,d,e,f]$ 과 $k = 2$ 에 대해, 주어진 문자열이 $P = [b,d,e]$ 라고 하자. 문자 b 의 L 에서의 index는 1이므로 $1 + k = 3$, 즉, L 에서의 index가 3인 문자인 d 가 cipher 문자이다. 마찬가지로 문자 d 에 대한 cipher 문자는 f 이다. 문자 e 인 경우 $h = 4$ 이므로 $h + k = 6 \geq n (=6)$ 이므로 $h + k - n = 0$, 즉, L 의 첫번째 문자 a 가 e 의 cipher 문자이다.

Cipher text를 de-cipher text로 바꾸는 것은 역의 과정을 통한다. 즉, 위 예에서 k 값을 -2 로 정하고 cipher 문자열 dfa 에 위와 같은 유사한 방법을 적용한다. 예를들어 d 의 L 에서의 index는 3이므로 $3 - 2 = 1$, 즉, cipher 문자 d 에 대한 de-cipher 문자는 b 이다. 그리고, a 의 L 에서의 index는 0이므로 $0 - 2 = -2$ 인데 값이 음수인 경우에는 0에서 cyclic하게 L 의 마지막에서 뒤로 두번째 문자라고 생각하면 된다. 즉, 우리의 예에서는 e 이다. 일반적으로 $h + k$ 가 음수인 경우 L 에서의 de-cipher 문자의 index는 $n - (h + k)$ 이다.

해결해야할 문제

문자 집합 S 를 영문 대문자 집합이라고 하고 L 을 이들을 알파벳 순서로 나열한 list라고 하자. 즉, $L = [A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z]$ 이다. 그리고, key 값 $k = 10$ 으로 정한다.

우리가 해결해야할 문제는 크기가 10인 일련의 cipher 문자열들이 주어졌을 때, 이를 decipher한 문자열들을 구하여 파일에 저장하는 프로그램을 작성하는 것이다.

입력

입력은 PHW02와 마찬가지로 `~.inc` 파일로 주어진다. 즉, 파일 `CSE3030_PHW03.inc`에 예를 들어 다음과 같은 형식으로 입력이 주어진다.

```
.data      ;;; key = 10
Num_Str    DWORD 2                ; Cipher 문자열의 개수(고정 값이 아님)
Cipher_Str BYTE "SVSUOKCCOW", 0   ; 정확히 10 개의 문자로 구성된 cipher text
          BYTE "RYGKBOIYEE", 0    ; 이러한 문자열이 Num_Str 개 만큼 반복
```

출력

입력에 포함된 각 문자열에 대한 de-cipher text를 구하여 파일에 저장한다. 여기서, 파일 이름은 **0snnnnnn_out.txt**이어야 한다(**nnnnnn**은 자신의 학번 뒤 6자리).

아래에 앞에서 예로 보인 입력의 문자열을 de-cipher한 결과를 저장한 파일 내용을 보인다.

```
ILIKEASSEM  
HOWAREYOUU
```

파일 생성 및 문자열 저장

파일 입출력에 관한 Irvine이 제공한 library 함수를 사용한다. 이 사용법은 강의에서 아직 제대로 다루지 않았으나, 학생들 스스로 help 파일 **c05_IrvineLibHelp.chm**를 참조하여 알 수 있다(이 파일은 예전에 배포하였음).

사용해야할 함수는 다음과 같이 C 언어에서의 파일 handling과 거의 유사하다.

```
CreateOutputFile      ; fopen과 동일  
WriteToFile           ; 문자열 하나를 저장할 수 있다  
CloseFile             ; fclose와 동일
```

참고로 도움말을 보면 file handle이라는 용어가 사용되는데, 이는 C 언어에서 file pointer와 동일하다고 생각하면 된다.

CreateOutputFile에서 file handle을 만들어 **eax**에 저장해 주고, 이를 **WriteToFile**, **CloseFile** 등을 호출할 때 **eax**에 저장하여 함수로 전달하여야 한다. 도움말에 예와 함께 설명되었으니 이를 참조하도록 하자. 도움말에 오류에 대한 언급이 있는데, 이는 처리 안해도 문제 없다.

제한사항

이 프로그램을 작성하는데 있어서, push, pop, call instruction 등을 포함하여 그 이전에 다룬 instructions, operators, directives 만을 사용하여야 한다. 조건문이 필요하지 않을까 생각할 수 있는데, 적절한 자료 선언, addressing 방법, stack operation, mov 관련 instruction, loop instruction, 약간의 트릭(?), 그리고 위에서 언급한 파일 관련 함수 호출만으로 충분히 작성할 수 있다. 추가 변수 및 초기화는 임의로 사용해도 무방하지만 가능한 적게 사용하도록 하자.

프로그램 작성 및 제출

파일 이름 : **snnnnnnHW03.asm** (여기서, **nnnnnn**은 자신의 학번 뒤 6 자리, **s**는 소문자).

위와 같은 .asm 파일 만을 사이버 캠퍼스 해당 과제함에 마감 전 제출 (late 없음).

주의 및 참고사항

- 다시 강조: push, pop, call 이후 다른 instructions 사용 불가. **예를 들어 이후에 다루는 cmp, test, conditional jump 등을 사용하면 0 점임.**
- 프로그램 크기(.lst 파일의 데이터 및 코드 크기 합)를 가능한 작게 작성하려 노력합니다.
- 추가 변수를 사용해도 무방하지만 가능한 레지스터를 사용하는 것이 크기를 줄이고 속도가 향상됩니다.
- 채점 시 어셈블 오류가 발생하면, 이유 불문 점수가 없습니다.
- 채점은 CSE3030_PHW03.inc의 변수 초기값을 바꿔 실행해볼 것입니다.
- 프로그램 초반부에 프로그램 작성자, 기능, 입력 그리고 출력 등을 comment로 기록하여야 하며 프로그램 중간 중간에 이해를 위하여 필요한 주석을 붙이려 노력하세요(남들이 봤을 때 이해가 가능하도록).
- 주석을 너무 많이 달아 오히려 이해하기 어렵게 한 경우는 감점합니다. 적절히 설명하세요.
- 출력 형식, 출력 파일 이름, 제출 파일 이름 등 위에서 요청한 것과 동일해야 합니다.
- 프로그램 복사는 철저히 점검할 것입니다. 복사로 판정되면 이유불문 쌍방 0점 처리합니다.