Assignment 03: Data

Date: 2024.09.23

Student ID:

Name: 김태훈

1. **정수, 식별자, 문자열 외에 추가로 정의할 수 있는 데이터의 형태(두 가지 이상 제시할 것)**

   추가로 정의할 수 있는 데이터는 먼저 실수(float)이 있다. 소수점을 포함하는 숫자 형태이다.

   그다음으로 boolean값이 있다. 이 형태는 'False', 'True'값만 가질 수 있다. 보통 'False'는 0에, 'True'는 1로 정의된다.

2. **제안한 데이터를 소스코드에 표현하는 방법**

   실수는 소스코드에 소수점을 기준으로 양옆에 숫자가 있다. 즉 3.14, 0.12, 12.34로 표시된다.

   Boolean은 소스코드에 True, true, False, false로 표시된다. True, true는 참을 의미하고 1이다. False, false는 거짓을 의미하고 0이다.

3. **제안한 데이터를 컴파일러의 타입으로 정의하는 방법**

```
typedef enum _Tag {
    TINT,
    TSTR,
    TFLOAT,
    TBOOL
} Tag;
```

```
typedef enum _boolean{
    FALSE,
    TRUE
    } boolean;

    typedef struct _Data {
        Tag tag;
        union {
            int ival;
            str sval;
            float fval;
            boolean bval;
        };
    } Data;
```

먼저 boolean은 enum을 사용하여 FALSE=0, TRUE=1로 하고[1], 소수는 c언어에 미리 정의되어있는 float으로 정의한다.

[1] 구현 방법

(0) Substr 구현[2]

```
(0)   str substr(str s, int start, int end){
(1)       str tmp = (str)malloc(sizeof(char)*(end-start+2));
(2)       strncpy(tmp,s+start,end-start+1);
(3)       tmp[end-start] = '\0';
(4)       return tmp;
(5)   }
```

Substr 함수는 기존 문자열에서 일부분을 잘라 새로운 문자열을 반환한다. 예를 들어 "HelloWorld" 문자열에 start=1, end=5값이 들어오면 "ell"을 반환한다.

(1) 정수와 소수

```
(6)   void processInput() {
(7)       Data d;
(8)       while (custom_gets() > 1) {
(9)           buf[strcspn(buf, "\n")] = '\0';
(10)          int i = 0;
(11)          while (i < strlen(buf)) {
(12)              if (isDigit(buf[i])) {
(13)                  int start = i;
(14)                  int isfloat=0;
(15)                  while (isDigit(buf[i]) || buf[i]=='.') {
(16)                      i++;
(17)                      if (buf[i]=='.') isfloat=1;
(18)                  }
(19)                  str tmp;
(20)                  if(!isfloat){
(21)                      d.tag = TINT;
(22)                      tmp = substr(buf,start,i);
(23)                      d.ival = atoi(tmp);
(24)                  }
(25)                  else{
(26)                      //float
(27)                      d.tag=TFLOAT;
(28)                      tmp = substr(buf,start,i);
(29)                      d.fval = atof(tmp);
(30)                  }
(31)                  datarray[datasz++] = d;
(32)                  free(tmp);
```

입력받은 한 라인의 문자열에서, 문자열이 숫자로 시작할 경우, 해당 문자부터 오른쪽으로 그 위치의 문자가 숫자나 소수점이 아닐 때 까지 이동하여 해당 위치까지 부분문자열을 만든다. 해당 부분문자열에 소수점이 포함되어 있으면 소수로, 아니면 정수로 분류하여 정수일경우 d.tag는 TINT, d.ival에는 문자열을 정수로 바꾸어 저장한다(atoi 함수 사용). 소수일경우 d.tag는 TFLOAT, d.fval에는 문자열을 소수로 바꾸어 저장한다(atof 함수 적용).

(2) 문자열과 Boolean

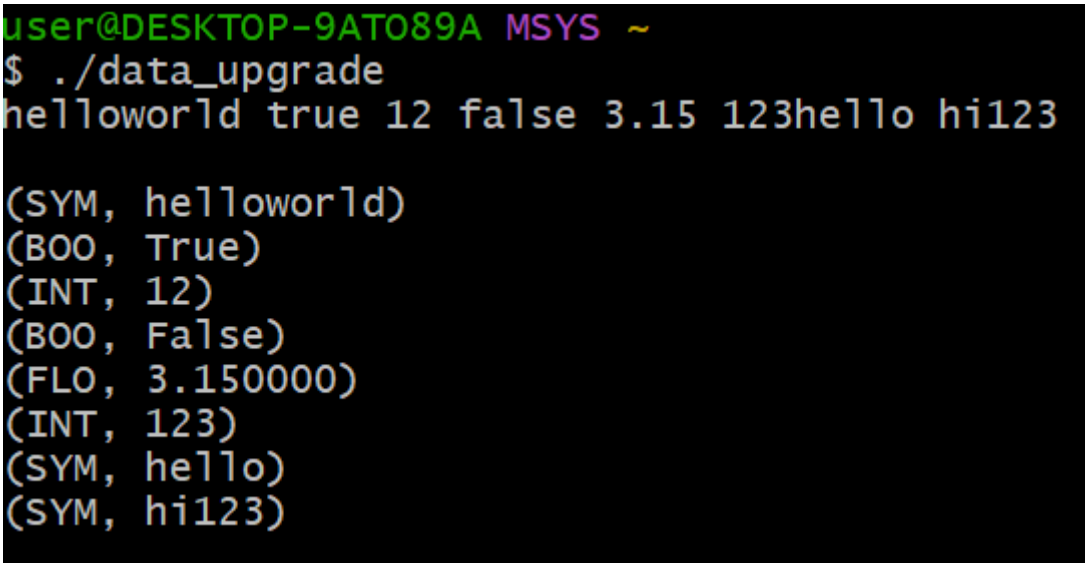
```
} else if (buf[i] == ' ') {
    i++;
} else {
    int start = i;
    while (buf[i] != ' ' && buf[i] != '\0') {
        i++;
    }

    str tmp;
    tmp = substr(buf,start,i);

    if(strlen(tmp) == 4 && (strncmp(tmp,"true",4) == 0 || strncmp(tmp,"True",4)==0)){
        d.tag = TBOOL;
        d.bval = TRUE;
        datarray[datasz++] = d;
        free(tmp);
    }
    else if(strlen(tmp)==5 && (strncmp(tmp,"false",5) == 0 || strncmp(tmp,"False",5)==0)){
        d.tag = TBOOL;
        d.bval = FALSE;
        datarray[datasz++] = d;
        free(tmp);
    }
    else{
        d.tag = TSTR;
        d.sval = substr(buf,start,i);
        datarray[datasz++] = d;

    }
```

입력받은 문자열에서 오른쪽으로 이동하면서 공백을 만나면 그냥 이동하고, 숫자와 공백을 제외한 문자열을 만나면 공백이나 null문자를 만날 때 까지 이동하여 부분문자열을 구성한다. 즉 "abcd12 efgh"에서 "abcd12"까지 부분 문자열을 구성한다.

이 부분 문자열의 크기가 4이고 "true"나 "True"라면 Boolean 타입(d.bval = TBOOL)에 값은 TRUE이다. 또한 크기가 5이고 값이 "false"나 "False"라면 Boolean 타입에 값은 FALSE이다.

그외에는 string으로 처리한다.

4. 작동화면



위 사진과 같이 문자, 정수, 실수, Boolean을 인식하여 출력한다.

참고문헌

[1] 남재윤 저. "C 언어 코딩 도장 개정판". 길벗. 2023.

https://dojang.io/mod/page/view.php?id=745 (visited at 2024.09.24)

[2] 강아지의 코딩공부 티스토리 c언어 substring 구현 : strncpy로 손쉽게 만들어 봅시다. https://codingdog.tistory.com/entry/c%EC%96%B8%EC%96%B4-substring-%EA%B5%AC%ED%98%84-strncpy%EB%A1%9C-%EC%86%90%EC%89%BD%EA%B2%8C-%EB%A7%8C%EB%93%A4%EC%96%B4-%EB%B4%85%EC%8B%9C%EB%8B%A4 (visited at 2024.09.24)