**２年 プログラミング基礎＆演習I（2024年7月12日（金））**

**1．1次元配列の初期化**

　次のように1次元配列を初期化することができる．多次元配列も初期化できるが，これは後期のポインタの授業で扱う．知りたい人は，補助教科書pp.97-100を参考にしてください．

init\_array.c

#include <stdio.h>

#define NUM 5

void disp\_array(int x[]);

int main(void)

{

int array[5] = {10, 20, 30, 40, 50};

int i;

for (i=0; i<NUM; i++) {

printf("array[%d] = %d\n", i, array[i]);

}

disp\_array(array);

return 0;

}

void disp\_array(int x[])

{

int i;

for (i=0; i<NUM; i++) {

printf("%d ", x[i]);

}

printf("\n");

}

**2．文字列配列の初期化**

　文字列配列を初期化するとき，文字列リテラルを与えることで簡単に初期化できる．しかし，“abc”のようなダブルクオートでくくった文字列リテラルは初期化にしか使えない．注意が必要である．

init\_string.c

#include <stdio.h>

#include <string.h>

#define NUM 5

void disp\_string(char s[]);

int main(void)

{

char str1[5+1] = {'a', 'b', 'c', 'd', 'e','\0'};

char str2[5+1] = {97, 98, 99, 100, 101,'\0'};

char str3[5+1] = "abcde";

char str4[] = "abcde";

char str5[3+1];

disp\_string(str1);

disp\_string(str2);

disp\_string(str3);

disp\_string(str4);

// 次のように，初期化以外で文字リテラルを与えることはできない．

//str5 = "abcde";

// 初期化以外で文字を与える場合，1文字ずつ与える．

str5[0] = 's';

str5[1] = 't';

str5[2] = 'u';

str5[3] = '\0';

disp\_string(str5);

// あるいはANSI Cライブラリ関数を使う．

strcpy(str5, "xyz");

disp\_string(str5);

return 0;

}

void disp\_string(char s[])

{

int i;

for (i=0; i<NUM; i++) {

printf("%c", s[i]);

}

printf("\n");

}

**第16回課題**

課題番号 No0712\_1.c

|  |
| --- |
| 文字列リテラル単語の文字数をカウントする関数を持つプログラムを作成せよ．ただし，文字の終端記号は文字数としてカウントしない．  例）  1．文字列としてprogrammingを初期化で与える．  2．この文字列をカウントする関数を作成する．  3．programmingという単語の文字数は11である． |

課題番号 No0712\_2.c

|  |
| --- |
| 英文の単語数をカウントするプログラムを作成しなさい．  例えば，英文  **Which do you think more exciting, soccer or baseball?**  の単語数は9個となる． |