

TI DSP, MCU, Xilinx Zynq FPGA 기반의 프로그래밍 전문가 과정

Homework : 5th day

강사 - 이상훈

학생 - 김형주

배열에 문자열을 입력 받고, 각 배열 요소가 짝수인 경우만을 출력하는 함수를 작성하라.

```
howard@ubuntu: ~/Mytest
1 #include <stdio.h>
2 void PrintStr(char* str,int size);
3 int main(void){
4     char str[33];
5     scanf("%s",str);
6
7     PrintStr(str,sizeof(str));
8
9     return 0;
10 }
11 void PrintStr(char* str,int size){
12     int i;
13     for(i=0;i<size;i++){
14         if(str[i]=='\0')
15             break;
16         if(!(str[i]%2))
17             printf("%c",str[i]);
18     }
19 }
```

```
howard@ubuntu:~/Mytest$ ./a.out
input str : ABCDEFGHIJKLMNOPQRSTUVWXYZ
output str : BDFHJLNPRTVXZ
howard@ubuntu:~/Mytest$
```

문자열을 입력받은 후, 각 문자에 대해서 아스키코드 값이 짝수인 경우만 출력

아래와 같은 숫자들이 배열에 들어 있다고 가정한다.

3, 77, 10, 7, 4, 9, 1, 8, 21, 33

이 요소들을 배열에 거꾸로 집어넣어보자.

```
howard@ubuntu: ~/Mytest
1 #include <stdio.h>
2 void arr_reverse(int* arr,int size);
3 int main(void){
4     int arr[]={3,77,10,7,4,9,1,8,21,33};
5     int size = sizeof(arr)/sizeof(int);
6     int i;
7     arr_reverse(arr,size);
8     for(i=0;i<size;i++){
9         printf("%d\n",arr[i]);
10    }
11
12    return 0;
13 }
14 void arr_reverse(int* arr,int size){
15     int temp1, temp2, i;
16     for(i=0;i<size/2;i++){
17         temp1 = arr[i];
18         temp2 = arr[size-1-i];
19         arr[i]=temp2;
20         arr[size-1-i]=temp1;
21     }
22
23 }
```

```
howard@ubuntu:~/Mytest$ ./a.out
33
21
8
1
9
4
7
10
77
3
```

Temp1과 temp2를 사용하여
중간 인덱스를 기준으로 대칭
되는 인덱스와 배열값 교환.

위의 숫자 3, 77, 10, 7, 4, 9, 1, 8, 21, 33에서
홀수 번째 요소의 합과 짝수 번째 요소의 합을 곱하시오.

```
howard@ubuntu: ~/Mytest
1 #include <stdio.h>
2 int arr_add(int* arr,int size,int even_odd_chk);
3 int main(void){
4     int arr[]={3,77,10,7,4,9,1,8,21,33};
5     int size = sizeof(arr)/sizeof(int);
6
7     printf("res = %d\n",arr_add(arr,size,0)*arr_add(arr,size,1));
8
9     return 0;
10 }
11 int arr_add(int* arr,int size,int even_odd_chk){
12     int i, sum = 0;
13     for(i=0;i<size;i++){
14         if(even_odd_chk==(i%2))
15             sum+=arr[i];
16     }
17     return sum;
18 }
```

```
howard@ubuntu:~/Mytest$ ./a.out
res = 5226
```

Even_odd_chk 변수가 1이면,
if(1==i%2)이므로 i가 홀수일
경우이고, even_odd_chk=0
이면, i가 짝수인 경우.

```

1 #include <stdio.h>
2
3 void AddMatrix(int (*mat)[3],int (*mat1)[3],int (*mat2)[3],int size_row,int size_col);
4 void SubMatrix(int (*mat)[3],int (*mat1)[3],int (*mat2)[3],int size_row,int size_col);
5 void MultipleMatrix(int (*mat)[3],int (*mat1)[3],int (*mat2)[3],int size_row,int size_col);
6
7 void PrintMatrix(int (*mat)[3],int size_row,int size_col);
8
9 int main(void){
10     int mat1[3][3]={1,2,3},{4,5,6},{7,8,9};
11     int mat2[3][3]={1,0,0},{0,1,0},{0,0,1};
12     int mat[3][3]={0};
13     int size_row = 3, size_col = 3;
14
15     //AddMatrix(mat,mat1,mat2,size_row,size_col);
16     PrintMatrix(mat,size_row,size_col);
17
18     //SubMatrix(mat,mat1,mat2,size_row,size_col);
19     PrintMatrix(mat,size_row,size_col);
20
21     MultipleMatrix(mat,mat1,mat2,size_row,size_col);
22     PrintMatrix(mat,size_row,size_col);
23
24     return 0;
25 }
26
27 void AddMatrix(int (*mat)[3],int (*mat1)[3],int (*mat2)[3],int size_row,int size_col){
28     int i, j;
29     for(i=0;i<size_row;i++){
30         for(j=0;j<size_col;j++){
31             mat[i][j]=mat1[i][j]+mat2[i][j];
32         }
33     }
34 }
35
36 void SubMatrix(int (*mat)[3],int (*mat1)[3],int (*mat2)[3],int size_row,int size_col){
37     int i, j;
38     for(i=0;i<size_row;i++){
39         for(j=0;j<size_col;j++){
40             mat[i][j]=mat1[i][j]-mat2[i][j];
41         }
42     }
43 }
44
45 void MultipleMatrix(int (*mat)[3],int (*mat1)[3],int (*mat2)[3],int size_row,int size_col){
46     int i, j, k;
47     for(i=0;i<size_row;i++){
48         for(j=0;j<size_col;j++){
49             for(k=0;k<size_col;k++){
50                 mat[i][j]+=mat1[i][k]*mat2[k][j];
51             }
52         }
53     }
54 }
55
56 void PrintMatrix(int (*mat)[3],int size_row,int size_col){
57     int i, j;
58     for(i=0;i<size_row;i++){
59         for(j=0;j<size_col;j++){
60             printf("%d ",mat[i][j]);
61         }
62         printf("\n");
63     }
64     printf("\n");
65 }

```

행렬의 곱셈, 덧셈, 나눗셈, 뺄셈에 대해 조사하시오.
숫자를 예로 들어서 계산도 해보시오.

```

howard@ubuntu:~/Mytest$ ./a.out
2 2 3
4 6 6
7 8 10

0 2 3
4 4 6
7 8 8

1 4 6
8 9 12
14 16 17

```

3 by 3 행렬에 대한 더하기, 빼기, 곱하기 함수 구현. 차후
2by3 행렬 X 3by4행렬 = 2by4
행렬 등 서로 다른 차원의 행렬 곱셈도 구현해 볼 예정.

우리는 예제에서 주소값을 교환하여 값을 변경하는 것을 해보았다. 그렇다면 변수 3개를 놓고, 이것에 대해서 무한 Loop를 돌면서 저글링을 해보자!

```
1 #include <stdio.h>
2
3 int main(void){
4     int i;
5     int n1 = 1, n2 = 2, n3 = 3;
6     int* temp = NULL;
7     int* n1_p = &n1;
8     int* n2_p = &n2;
9     int* n3_p = &n3;
10    int** n_p_p = &n1_p;
11    i=0;
12    while(i++<10){
13
14        printf("*n1_p=%d\n*n2_p=%d\n*n3_p=%d\n\n",*n1_p,*n2_p,*n3_p);
15        temp = *n_p_p;
16        *n_p_p = n2_p;
17        n2_p = n3_p;
18        n3_p = temp;
19    }
20
21    return 0;
22 }
23 }
```

```
*n1_p=1
*n2_p=2
*n3_p=3
```

```
*n1_p=2
*n2_p=3
*n3_p=1
```

```
*n1_p=3
*n2_p=1
*n3_p=2
```

```
*n1_p=1
*n2_p=2
*n3_p=3
```

무한 루프 내 delay하는 방법을 몰라 i<10이라는 조건을 걸었다.

삼각형의 넓이를 구하라.

```
1 #include <stdio.h>
2 #include <math.h>
3 double triangleArea(int x,int h);
4 int main(void){
5     int x = 4, h = 8;
6     double area;
7
8     printf("삼각형의 넓이 (밑변x높이/2) : %.2lf\n",triangleArea(x,h));
9
10
11     return 0;
12 }
13 double triangleArea(int x,int h){
14     return x*h/2;
15 }
```

```
howard@ubuntu:~/Mytest$ ./a.out
삼각형의 넓이 (밑변x높이/2) : 16.00
```

삼각형의 넓이 = 밑변x높이/2

2 by 2 행렬의 곱셈을 계산할 수 있는 프로그램을 만드시오.

```
1 #include <stdio.h>
2 void multipleMat(int (*mat1)[2],int (*mat2)[2]);
3 int main(void){
4     int mat1[2][2]={{1,1},{2,2}};
5     int mat2[2][2]={{2,2},{0,0}};
6
7     multipleMat(mat1,mat2);
8
9     return 0;
10 }
11 void multipleMat(int (*mat1)[2],int (*mat2)[2]){
12     int arr[2][2]={0};
13     int i,j,k;
14
15     for(i=0;i<2;i++){
16         for(j=0;j<2;j++){
17             for(k=0;k<2;k++){
18                 arr[i][j]+=mat1[i][k]*mat2[k][j];
19             }
20         }
21     }
22     printf("mat1 :\n");
23     for(i=0;i<2;i++){
24         for(j=0;j<2;j++){
25             printf("%d\t",mat1[i][j]);
26             printf("\n");
27         }
28     }
29     printf("mat2 :\n");
30     for(i=0;i<2;i++){
31         for(j=0;j<2;j++){
32             printf("%d\t",mat2[i][j]);
33             printf("\n");
34         }
35     }
36     printf("mat1xmat2 :\n");
37     for(i=0;i<2;i++){
38         for(j=0;j<2;j++){
39             printf("%d\t",arr[i][j]);
40             printf("\n");
41         }
42     }
43 }
```

```
mat1 :
1      1
2      2
mat2 :
2      2
0      0
mat1xmat2 :
2      2
4      4
```