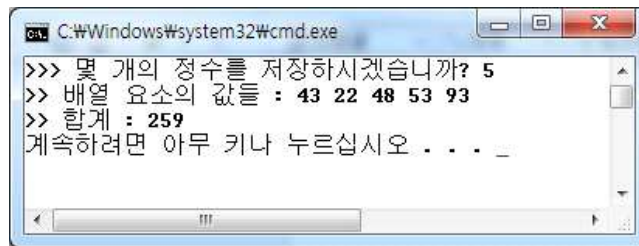
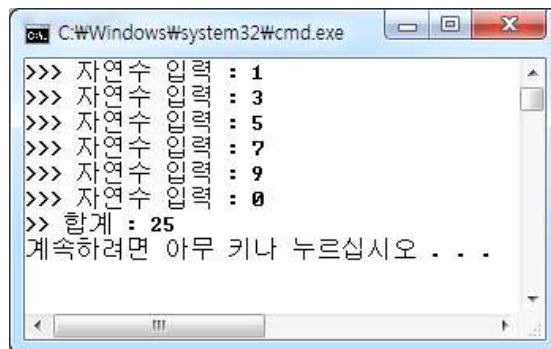


- 문제1 : 사용자로부터 먼저 저장할 정수의 개수를 입력받는다. 그리고 int 포인터를 사용하여 해당 개수만큼의 int 배열을 만들고 rand 함수를 사용하여 1~100 사이의 임의의 값으로 채워 보라. 마지막으로 해당 배열의 모든 요소의 값과 합산한 결과를 출력하라.



- 문제2 : 사용자로부터 자연수를 하나씩 입력받아 변수에 저장하다가 0 이하의 값이 입력되면 입력을 멈춘다. 그리고 나서 입력받은 모든 자연수의 값을 합산한 결과를 출력하라. 그런데 문제는 사용자가 몇 개의 자연수를 입력할지 알 수 없다는 것이다. 따라서 자연수 하나를 추가로 입력할 때마다 malloc 함수를 사용하여 동적으로 할당하는 배열의 크기를 하나씩 증가시키거나(물론 기존 값들을 보존할 수 있어야 한다), realloc 함수를 사용하여 배열의 크기를 하나씩 증가시킬 수 있어야 한다. (아마도 realloc 함수를 사용하면 훨씬 편할 것이다.)



□ 코드

< 1번 문제 >

```
#include <stdio.h>
#include <stdlib.h>
```

```
int main(void) {
    int size;
    int *arr;
```

```

int sum = 0;

printf(">>> 몇 개의 정수를 저장하시겠습니까? : ");
scanf("%d", &size);

arr = (int*)malloc(sizeof(int) * size);

for (int i = 0; i < size; i++) {
    arr[i] = rand() % 100 + 1;
}
printf(">>> 배열 요소의 값들 : ");

for (int i = 0; i < size; i++)
{
    printf("%d ", arr[i]);
    sum += arr[i];
}

printf("\n>>> 배열 요소의 합 : %d ", sum);
free(arr);

return 0;
}

```

< 2번 문제 >

```

#include <stdio.h>
#include <stdlib.h>

```

```

int main(void) {
    int input = 1;
    int sum = 0;
    int size = 1;

    int *ary = (int*)malloc(sizeof(int) * 1);

    while (1) {
        printf(">>> 자연수 입력 : ");
        scanf("%d", &input);

        if (input == 0)          // input값이 0이라면
            break;
    }
}

```

```

        sum += input;
        ary = (int*)realloc(ary, sizeof(int) * ++size);
        ary[size - 1] = input;
    }
    printf(">> 합계 : %d", sum);
    return 0;
}

```

□ 실행 예

< 1번 문제 >

Microsoft Visual Studio 디버그 콘솔

```

>>> 몇 개의 정수를 저장하시겠습니까? : 3
>>> 배열 요소의 값들 : 42 68 35
>>> 배열 요소의 합 : 145

```

Microsoft Visual Studio 디버그 콘솔

```

>>> 몇 개의 정수를 저장하시겠습니까? : 5
>>> 배열 요소의 값들 : 42 68 35 1 70
>>> 배열 요소의 합 : 216

```

< 2번 문제 >

Microsoft Visual Studio 디버그 콘솔

```

>>> 자연수 입력 : 1
>>> 자연수 입력 : 2
>>> 자연수 입력 : 3
>>> 자연수 입력 : 4
>>> 자연수 입력 : 5
>>> 자연수 입력 : 0
>> 합계 : 15

```

Microsoft Visual Studio 디버그 콘솔

```

>>> 자연수 입력 : 8
>>> 자연수 입력 : 5
>>> 자연수 입력 : 6
>>> 자연수 입력 : 1
>>> 자연수 입력 : 3
>>> 자연수 입력 : 8
>>> 자연수 입력 : 4
>>> 자연수 입력 : -8
>>> 자연수 입력 : 0
>> 합계 : 27

```