

# 동적 메모리 할당

C Language dynamic memory allocation

SCP  
1학년 심재훈

# ● Contents

- 01 동적 메모리 할당을 공부한 이유
- 02 동적 메모리 할당이란?
- 03 정적 메모리 vs 동적 메모리
- 04 동적 메모리 할당 방법
- 05 동적 메모리 할당 실습

# 01 동적 메모리 할당을 공부한 이유

## Caesar 암호



```
// 암호화 하는 함수
void ency() {
    int ekey, i;
    char m[100]={0};
    char enc[100] = {0};
    printf("평문을 입력해주세요 : ");
    scanf_s("%[^\n]s", &m, 100); // 평문 입력 받기
```

```
// 복호화 하는 함수
void decy() {
    int dkey, j;
    char denc[100] = { 0 };
    char dec[100] = { 0 };
    printf("암호문을 입력해주세요 : ");
    scanf_s("%[^\n]s", &denc, 100); // 암호문 입력 받기
```

## 02 동적 메모리 할당이란?

프로그램 실행 도중에 사용할 메모리 공간(힙)을 할당받는 것



# 03 정적 메모리 vs 동적 메모리

## 메모리 구조

Low memory

코드

데이터

힙

스택

High memory

Ex) arr[5];

### 정적 메모리 할당

메모리 할당	실행 전
메모리 크기	제한적 / 실행 중 조절 불가능
메모리 관리	자동 관리
메모리 할당 해제	자동 반납
장점	메모리 누수 걱정 x
단점	메모리 공간 낭비 발생

### 동적 메모리 할당

메모리 할당	실행 중
메모리 크기	실행 중 유동적으로 조절 가능
메모리 관리	직접 관리
메모리 할당 해제	직접 반납
장점	메모리 공간 낭비 x
단점	할당 해제를 직접 해야함

# 04 동적 메모리 할당 방법

Memory ALLOcation의 줄임말

사용 언어 : C언어  
헤더파일 - stdlib.h

C++

메모리 할당 함수 - malloc() / new()  
메모리 해제 함수 - free() / delete()

malloc() 사용 예시

```
int *p = (int*)malloc(sizeof(int));
```

free() 사용 예시

```
free(p);
```

함수	기능
<code>void * malloc ( size_t size );</code>	size 바이트의 메모리를 힙에서 할당하여 반환한다.
<code>void * calloc ( size_t num, size_t size );</code>	(num * size) 바이트의 메모리를 힙에서 할당하고 포인터값을 반환한다.
<code>void * realloc ( void * ptr, size_t size );</code>	ptr이 가리키는 메모리를 size 바이트만큼 힙에서 재할당하여 반환한다.
<code>void free ( void * ptr );</code>	ptr이 가리키는 메모리를 해제한다. 해제 전까지 계속 존재하므로 필요없으면 이 함수에 의해 해제해야 한다.

출처 : 위키백과 ([https://ko.wikipedia.org/wiki/%EB%8F%99%EC%A0%81\\_%EB%A9%94%EB%AA%A8%EB%A6%AC\\_%ED%95%A0%EB%8B%B9](https://ko.wikipedia.org/wiki/%EB%8F%99%EC%A0%81_%EB%A9%94%EB%AA%A8%EB%A6%AC_%ED%95%A0%EB%8B%B9))

# 05 동적 메모리 할당 실습

사용 언어 : C언어 / 과목의 평균 점수를 구하는 프로그램 ( 과목 개수만큼 동적할당 )

```
1 #include <stdio.h>
2 #include <stdlib.h> // malloc(), free()
3
4 int main()
5 {
6     int subject_count; // 과목 개수 변수
7     int *subject_count_arr; // 동적메모리할당 포인터 변수
8     int increase, sum=0; // 증가값, 합계 변수
9
10    //과목 개수 입력
11    printf("과목 개수 입력 : ");
12    scanf_s("%d", &subject_count);
13
14    // 과목 개수가 1보다 미만이면 종료
15    if (subject_count < 1)
16        return 0;
17
18    subject_count_arr = (int*)malloc(sizeof(int) * subject_count); // 동적메모리할당
19
20    // 동적메모리할당 실패시 종료
21    if (subject_count_arr == NULL)
22        return 0;
23
24    // 과목 점수 입력
25    for (increase = 0; increase < subject_count; increase++)
26    {
27        printf("%d번째 과목 점수 입력 : ");
28        scanf_s("%d", &subject_count_arr[increase]);
29    }
30
31    // 과목 점수 합계 계산
32    for (increase = 0; increase < subject_count; increase++)
33        sum += subject_count_arr[increase];
34
35    printf("평균 : %d점", sum / subject_count); // 평균 점수 출력
36    free(subject_count_arr); // 동적메모리할당 해제
37    return 0;
38 }
```

#include <stdlib.h> // malloc(), free()

int subject\_count; // 과목 개수 변수

int \*subject\_count\_arr; // 동적메모리할당 포인터 변수 = 점수를 입력받을 배열

int increase, sum=0; // 증가값, 합계 변수

# 05 동적 메모리 할당 실습

사용 언어 : C언어 / 과목의 평균 점수를 구하는 프로그램 ( 과목 개수만큼 동적할당 )

```
1 #include <stdio.h>
2 #include <stdlib.h> // malloc(), free()
3
4 int main()
5 {
6     int subject_count; // 과목 개수 변수
7     int *subject_count_arr; // 동적메모리할당 포인터 변수
8     int increase, sum=0; // 증가값, 합계 변수
9
10    //과목 개수 입력
11    printf("과목 개수 입력 : ");
12    scanf_s("%d", &subject_count);
13
14    // 과목 개수가 1보다 미만이면 종료
15    if (subject_count < 1)
16        return 0;
17
18    subject_count_arr = (int*)malloc(sizeof(int) * subject_count); // 동적메모리할당
19
20    // 동적메모리할당 실패시 종료
21    if (subject_count_arr == NULL)
22        return 0;
23
24    // 과목 점수 입력
25    for (increase = 0; increase < subject_count; increase++)
26    {
27        printf("%d번째 과목 점수 입력 : ", increase+1);
28        scanf_s("%d", &subject_count_arr[increase]);
29    }
30
31    // 과목 점수 합계 계산
32    for (increase = 0; increase < subject_count; increase++)
33        sum += subject_count_arr[increase];
34
35    printf("평균 : %d점", sum / subject_count); // 평균 점수 출력
36    free(subject_count_arr); // 동적메모리할당 해제
37    return 0;
38 }
```

//과목 개수 입력

```
printf("과목 개수 입력 : ");
scanf_s("%d", &subject_count);
```


```
// 과목 개수가 1보다 미만이면 종료
if (subject_count < 1)
    return 0;
```



# 05 동적 메모리 할당 실습

사용 언어 : C언어 / 과목의 평균 점수를 구하는 프로그램 ( 과목 개수만큼 동적할당 )

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 subject_count_arr = (int*)malloc(sizeof(int) * subject_count); // 동적메모리할당
4 int main()
5 {
6     // subject_count_arr[subject_count];
7     int
8     int
9     int
10    // 동적메모리할당 실패시 종료
11    printf("동적메모리할당 실패시 종료\n");
12    if (subject_count_arr == NULL)
13    {
14        return 0;
15    }
16    return 0;
17
18    subject_count_arr = (int*)malloc(sizeof(int) * subject_count); // 동적메모리할당
19
20    // 동적메모리할당 실패시 종료
21    if (subject_count_arr == NULL)
22    {
23        return 0;
24    }
25
26    // 과목 점수 입력
27    for (increase = 0; increase < subject_count; increase++)
28    {
29        printf("%d번째 과목 점수 입력 : ", increase+1);
30        scanf_s("%d", &subject_count_arr[increase]);
31    }
32
33    // 과목 점수 합계 계산
34    for (increase = 0; increase < subject_count; increase++)
35        sum += subject_count_arr[increase];
36
37    printf("평균 : %d점", sum / subject_count); // 평균 점수 출력
38    free(subject_count_arr); // 동적메모리할당 해제
39    return 0;
40 }
```



# 05 동적 메모리 할당 실습

사용 언어 : C언어 / 과목의 평균 점수를 구하는 프로그램 ( 과목 개수만큼 동적할당 )

```
1 #include <stdio.h>
2 #include <stdlib.h> // malloc(), free()
3
4 int main()
5 {
6     int subject_count; // 과목 개수 변수
7     int *subject_count_arr; // 동적메모리할당 포인터 변수
8     int increase, sum=0; // 증가값, 합계 변수
9
10    //과목 개수 입력
11    printf("과목 개수 입력 : ");
12    scanf_s("%d", &subject_count);
13
14    // 과목 개수가 1보다 미만이면 종료
15    if (subject_count < 1)
16        return 0;
17
18    subject_count_arr = (int*)malloc(sizeof(int) * subject_count); // 동적메모리할당
19
20    // 동적메모리할당 실패시 종료
21    if (subject_count_arr == NULL)
22        return 0;
23
24    // 과목 점수 입력
25    for (increase = 0; increase < subject_count; increase++)
26    {
27        printf("%d번째 과목 점수 입력 : ", increase+1);
28        scanf_s("%d", &subject_count_arr[increase]);
29    }
30
31    // 과목 점수 합계 계산
32    for (increase = 0; increase < subject_count; increase++)
33        sum += subject_count_arr[increase];
34
35    printf("평균 : %d점", sum / subject_count); // 평균 점수 출력
36    free(subject_count_arr); // 동적메모리할당 해제
37    return 0;
38 }
```

// 과목 점수 입력

```
for (increase = 0; increase < subject_count; increase++)
{
    printf("%d번째 과목 점수 입력 : ", increase+1);
    scanf_s("%d", &subject_count_arr[increase]);
}
```

// 과목 점수 합계 계산

```
for (increase = 0; increase < subject_count; increase++)
    sum += subject_count_arr[increase];
```

# 05 동적 메모리 할당 실습

사용 언어 : C언어 / 과목의 평균 점수를 구하는 프로그램 ( 과목 개수만큼 동적할당 )

```
1 #include <stdio.h>
2 #include <stdlib.h> // malloc(), free()
3
4 int main()
5 {
6     int subject_count; // 과목 개수 변수
7     int *subject_count_arr; // 동적 메모리할당 포인터 변수
8     int increase, sum=0; // 증가값, 합계 변수
9
10    //과목 개수 입력
11    printf("과목 개수 입력 : ");
12    scanf_s("%d", &subject_count);
13
14    // 과목 개수가 1보다 미만이면 종료
15    if (subject_count < 1)
16        return 0;
17
18    subject_count_arr = (int*)malloc(sizeof(int) * subject_count); // 동적메모리할당
19
20    // 동적메모리할당 실패시 종료
21    if (subject_count_arr == NULL)
22        return 0;
23
24    // 과목 점수 입력
25    for (increase = 0; increase < subject_count; increase++)
26    {
27        printf("%d번째 과목 점수 입력 : ", increase+1);
28        scanf_s("%d", &subject_count_arr[increase]);
29    }
30
31    // 과목 점수 합계 계산
32    for (increase = 0; increase < subject_count; increase++)
33        sum += subject_count_arr[increase];
34
35    printf("평균 : %d점", sum / subject_count); // 평균 점수 출력
36    free(subject_count_arr); // 동적메모리할당 해제
37    return 0;
38 }
```

printf("평균 : %d점", sum / subject\_count); // 평균 점수 출력

free(subject\_count\_arr); // 동적메모리할당 해제



# 05 동적 메모리 할당 실습

사용 언어 : C언어 / 과목의 평균 점수를 구하는 프로그램 ( 과목 개수만큼 동적할당 )

```
1 #include <stdio.h>
2 #include <stdlib.h> // malloc(), free()
3
4 int main()
5 {
6     int subject_count; // 과목 개수 변수
7     int *subject_count_arr; // 동적메모리할당 포인터 변수
8     int increase, sum=0; // 증가값, 합계 변수
9
10    //과목 개수 입력
11    printf("과목 개수 입력 : ");
12    scanf_s("%d", &subject_count);
13
14    // 과목 개수가 1보다 미만이면 종료
15    if (subject_count < 1)
16        return 0;
17
18    subject_count_arr = (int*)malloc(sizeof(int) * subject_count); // 동적메모리할당
19
20    // 동적메모리할당 실패시 종료
21    if (subject_count_arr == NULL)
22        return 0;
23
24    // 과목 점수 입력
25    for (increase = 0; increase < subject_count; increase++)
26    {
27        printf("%d번째 과목 점수 입력 : ", increase+1);
28        scanf_s("%d", &subject_count_arr[increase]);
29    }
30
31    // 과목 점수 합계 계산
32    for (increase = 0; increase < subject_count; increase++)
33        sum += subject_count_arr[increase];
34
35    printf("평균 : %d점", sum / subject_count); // 평균 점수 출력
36    free(subject_count_arr); // 동적메모리할당 해제
37    return 0;
38 }
```

## 실행 결과

```
과목 개수 입력 : 5
1번째 과목 점수 입력 : 100
2번째 과목 점수 입력 : 40
3번째 과목 점수 입력 : 60
4번째 과목 점수 입력 : 80
5번째 과목 점수 입력 : 20
평균 : 60점
```



**Q & A**



# 감사합니다.

Thank You For Listening