

# **교**학습내용

- ♀ 12.3 동적 메모리 사용 예제 (2)
- **♀** 12.4 기타 동적 메모리 할당 함수



### **할 학습목표**

- ♀ 12.3 예제를 통해 동적 메모리 할당 사용법을 익힌다.
- 12.4 기타 동적 메모리 할당 함수를 학습한다.





C프로그래밍및실습





- 1. 동적 할당 개요
- 2. 동적 메모리 사용 절
- 3. 동적 메모리 사용 예제 (2)
- 4. 기타 동적 메모리 할당 함수



#### 🔳 3) 동적 메모리 할당을 사용한 문자열 처리 프로그램

- 길이가 다른 여러 개의 문자열을 효율적으로 저장하기 위해
- 다음과 같은 방식으로 동적 메모리 할당 이용
  - ① 문자열을 입력 받기 위한 충분한 크기의 문자 배열을 선언 (정적 할당)
  - ② ①의 문자 배열에 문자열을 입력 받음
  - ③ ②에서 입력된 문자열의 길이를 계산하여 그 크기에 맞게 메모리를 동적으로 할당 받음(널 문자 포함)
  - ④ 할당 받은 메모리 공간에 ①의 문자 배열의 문자열을 복사함



#### 🔳 3) 동적 메모리 할당을 사용한 문자열 처리 프로그램

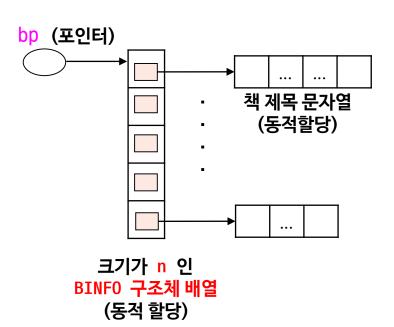
● 문제 : 사용자로부터 <mark>책의 개수 n을 입력 받아</mark>, n개의 책 제목을 저장하는 프로그램을 작성하시오.

#### ✓ 사용할 메모리 구조

✓ 책 하나의 정보를 저장하는 구조체

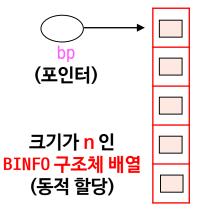
```
typedef struct book_title {
  char *title; // 책 제목(포인터)
          // 기타 책 정보(필요 시)
 BINFO;
```

✓ 책의 개수를 모르므로 동적 구조체 배열 사용



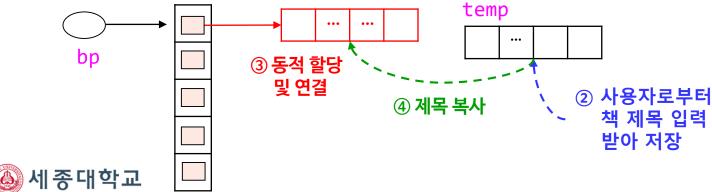


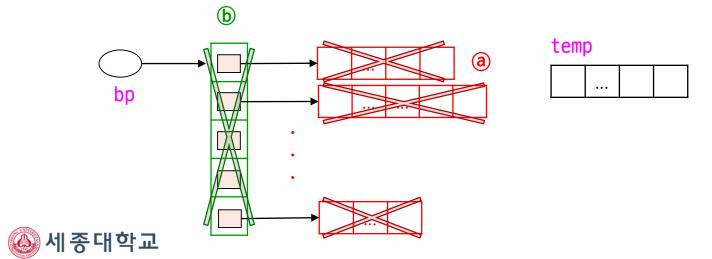
```
BINFO *bp = NULL;
int n, i, len;
char temp[100];
             // ① 문자열을 입력 받기 위한
                   문자 배열을 선언
scanf("%d", &n); // 책 개수 입력
getchar(); // 개행 문자 버리기
// 책 배열 동적 할당 및 동적 할당 오류 검사
bp = (BINFO *)malloc( n*sizeof(BINFO) );
if (bp == NULL) { ... }
(다음 슬라이드에 계속)
```













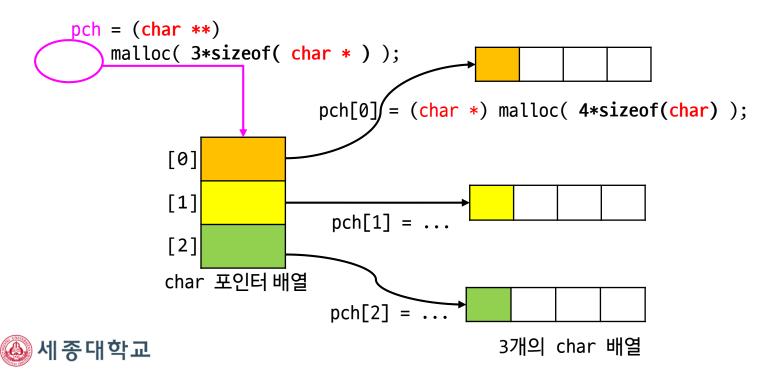
### 💷 4) 동적 메모리 할당을 사용하여 <mark>2차원 배열</mark> 할당하는 프로그램

- 문제 : char array [3] [4] 라는 2차원 배열을 동적으로 할당하기
- 착각하기 쉬운 잘못된 방법
  - ✓ char \*pch = (char \*) malloc( 3\*4\*sizeof(char) );
    - → 사실상 malloc(12);호출
  - ✓ 2차원 배열 할당이 아니라, 크기가 12인 1차원 배열 할당
- 해결방법: 포인터 배열 사용(다음 슬라이드 참조)
  - ✓ 다른 다차원 배열도 동일한 원리 적용: 2차원 정수 배열, 3차원 배열 등



#### 🗊 4) 동적 메모리 할당을 사용하여 <mark>2차원 배열</mark> 할당하는 프로그램

● 해결방법: 포인터 배열 사용! (메모리 그림 참조)



```
int i; char **pch;
                                         // 이중 포인터 선언
pch = (char **)malloc( 3*sizeof(char *) ); // 포인터 배열 할당
for (i=0; i<3; i++)
  pch[i] = (char *)malloc( 4*sizeof(char) ); // 1차원 배열 할당
strcpy(pch[0], "aaa");
strcpy(pch[1], "bbb");
pch[2][0] = '\0'; // 2차원 배열처럼 사용 (예제 9.10 참조)
for (i=0; i<3; i++)
  puts(pch[i]);
for (i=0; i<3; i++)
  free(pch[i]);
                        // 1차원 배열 메모리 해제
free(pch);
                         // 포인터 배열 메모리 해제
```



C프로그래밍및실습



- 1. 동적 할당 개요
- 2. 동적 메모리 사용 절
- 3. 동적 메모리 사용 예제 (2)
- 4. 기타 동적 메모리 할당 함수

#### 4. 기타 동적 메모리 할당 함수

## alloc() 함수

함수 원형	<pre>void * calloc(unsigned int num, unsigned int size);</pre>		
함수 인자	num	동적으로 할당 받을 원소의 개수	
	size	원소 한 개의 크기 (바이트 단위)	
반환 값	<ul> <li>✓ (num*size) 바이트 수 만큼 할당하고, 할당된 메모리를 모두 0으로 초기화 후 시작 위치 반환</li> <li>✓ 실패하면 → NULL 반환</li> </ul>		



#### 4. 기타 동적 메모리 할당 함수

# realloc() 함수

함수 원형	<pre>void * realloc(void *ptr, unsigned int size);</pre>			
함수 인자	ptr	확장 크기를 변경할 메모리의 시작 주소		
	new_size	변경 후 메모리의 전체 크기 (바이트 단위)		
반환 값	<ul> <li>✓ ptr이 가리키는 메모리의 크기를 new_size 바이트 크기로 조정하고, 조정된 메모리의 시작 위치 반환</li> <li>✓ 실패하면 → NULL 반환</li> </ul>			

```
(사용 예)
char *p = (char *) malloc( 5 ); ⇒ 5 바이트 만큼 공간 할당
p = (char *) realloc( p, 10 ); ⇒ 10 바이트로 공간 크기 변경
```

✓ 주의!! 기존 위치에서 새로운 크기를 확보할 수 없으면, 기존 위치의 공간을 해제하고 <mark>새로운 위치에 공간</mark> 할당



### 학습정리

- 동적 할당을 다단계로 적용하여 복잡한 구조의 동적 메모리를 사용할 수 있음(예제 3)
- **2차원 동적 배열**은 포인터 배열을 사용하여 두 단계의 동적 할당으로 표현함(예제 4)
- calloc() 함수는 malloc() 함수와 유사하게 메모리를 할당해 주는데, 추가로 할당 받은 메모리 공간을 0으로 초기화 함
- realloc() 함수는 할당 받은 메모리 공간의 크기를 변경하고자 할 때 사용

