



DATA STRUCTURES AND ITS APPLICATIONS

Vandana M L

Department of Computer Science and Engineering

DATA STRUCTURES AND ITS APPLICATIONS

Memory Allocation

Vandana M L

Department of Computer Science and Engineering

- Static Memory Allocation
- Dynamic Memory Allocation

- allocated by the compiler.
- Exact size and type of memory must be known at compile time
- Memory is allocated in stack area

```
int b;
```

```
int c[10] ;
```

- Memory allocated can not be altered during run time as it is allocated during compile time
- This may lead to under utilization or over utilization of memory
- Memory can not be deleted explicitly only contents can be overwritten
- Useful only when data size is fixed and known before processing

- Dynamic memory allocation is used to obtain and release memory during program execution.
- It operates at a low-level
- Memory Management functions are used for allocating and deallocating memory during execution of program
- These functions are defined in “stdlib.h”

Dynamic Memory Allocation Functions:

- Allocate memory - malloc(), calloc(), and realloc()
- Free memory - free()

To allocate memory use

```
void *malloc(size_t size);
```

- Takes number of bytes to allocate as argument.
- Use sizeof to determine the size of a type.
- Returns pointer of type void *. A void pointer may be assigned to any pointer.
- If no memory available, returns NULL.

To allocate space for 100 integers:

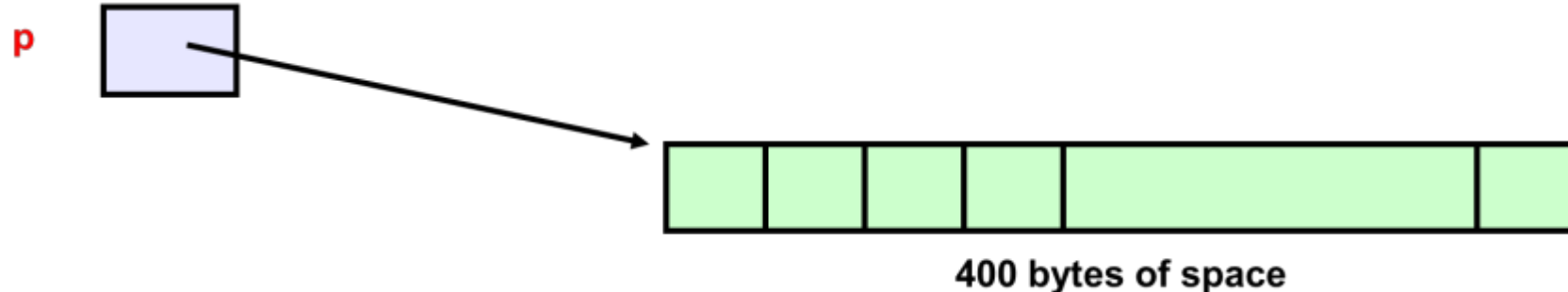
```
int *p;
```

```
if ((p = (int*)malloc(100 * sizeof(int))) == NULL){
```

```
    printf("out of  
memory\n");  
    exit();
```

```
}
```

- Note we cast the return value to int*.
- Note we also check if the function returns NULL.



Dynamic Memory Allocation Functions: malloc()

- `cptr = (char *) malloc (20);`

Allocates 20 bytes of space for the pointer `cptr` of type `char`

- `sptr = (struct stud *) malloc(10*sizeof(struct stud));`

Allocates space for a structure array of 10 elements. `sptr` points to a structure element of type `struct stud`

Always use sizeof operator to find number of bytes for a data type, as it can vary from machine to machine

Dynamic Memory Allocation Functions: malloc()

- **malloc** always allocates a block of contiguous bytes
 - The allocation can fail if sufficient contiguous memory space is not available
 - If it fails, **malloc** returns **NULL**

```
if ((p = (int *) malloc(100 * sizeof(int))) == NULL)
{
    printf ("\n Memory cannot be allocated");
    exit();
}
```

The n integers allocated can be accessed as ***p, *(p+1), *(p+2),..., *(p+n-1)** or just as **p[0], p[1], p[2], ...,p[n-1]**

To release allocated memory use

`free(ptrvariable)`

- Deallocates memory allocated by malloc().
- Takes a pointer as an argument.

e.g

. `free(newPtr);`



Memory
Leak



Dangling
pointer

Similar to malloc(),

But allocated memory space are zero by default..

calloc() requires two arguments –

```
void *calloc(size_t nitem, size_t size);
```

Example

```
int *p;
```

```
p=(int*)calloc(100,sizeof(int));
```

returns a void pointer if the memory allocation is successful,
else it'll return a NULL pointer.

Reallocate a block

Two arguments

- Pointer to the already allocated block
- Size of new block

```
int *ip;
```

```
ip = (int*)malloc(100 * sizeof(int));
```

```
...
```

```
/* need twice as much space */
```

```
ip = (int*)realloc(ip, 200 * sizeof(int));
```

Memory Allocation

- Static Memory allocation
- Dynamic memory allocation

Apply the concepts to implement C program for the following problem statement

- Multiply two matrices . Allocate the memory for the matrices dynamically

1. Which of the following is true about static memory allocation?

- a) Memory is allocated during runtime.
- b) The size of memory block can be altered during execution.
- c) Memory is allocated during compile time and remains fixed.
- d) It uses malloc() and free() functions.

1. Which of the following is true about static memory allocation?

- a) Memory is allocated during runtime.
- b) The size of memory block can be altered during execution.
- c) Memory is allocated during compile time and remains fixed.
- d) It uses malloc() and free() functions.

2. In C, which of the following dynamic allocation functions returns memory that is not initialized?

- a) malloc()
- b) calloc()
- c) realloc()
- d) free()

2. In C, which of the following dynamic allocation functions returns memory that is not initialized?

- a) `malloc()`
- b) `calloc()`
- c) `realloc()`
- d) `free()`

3. Which of the following statements is false regarding dynamic memory allocation?

- a) It is performed at runtime.
- b) It uses heap memory.
- c) The size of memory block cannot be changed once allocated.
- d) `realloc()` can resize an already allocated block.

3. Which of the following statements is false regarding dynamic memory allocation?

- a) It is performed at runtime.
- b) It uses heap memory.
- c) The size of memory block cannot be changed once allocated.
- d) realloc() can resize an already allocated block.

4. What will happen if free() is not called after using dynamically allocated memory?

- a) The program will not compile.
- b) Memory leak will occur.
- c) The stack will overflow.
- d) The data will be automatically deleted.

4. What will happen if free() is not called after using dynamically allocated memory?

- a) The program will not compile.
- b) Memory leak will occur.
- c) The stack will overflow.
- d) The data will be automatically deleted.

5. Which of the following is incorrect?

- a) Static allocation uses stack, dynamic uses heap.
- b) Static allocation is determined at compile time, dynamic at runtime.
- c) Static memory can be resized using realloc().
- d) Dynamic memory requires explicit allocation using malloc/calloc.

5. Which of the following is incorrect?

- a) Static allocation uses stack, dynamic uses heap.
- b) Static allocation is determined at compile time, dynamic at runtime.
- c) Static memory can be resized using realloc().
- d) Dynamic memory requires explicit allocation using malloc/calloc.



THANK YOU

Vandana M L

Department of Computer Science & Engineering

vandanamd@pes.edu

+91 7411716615