Assignment - 18

- what is difference between static memory 9.1 allocation and dynamic memory allocations,
 - · Static memory means the amount of memory to be given to the program done at compile time.
 - Dynamic memory means the amount of memory to be given to program is done at run time
 - of memory done i at compile time a o Static memory allocation is considered as early binding!
 - Dynamic memory allocation is considered as appreach some time memory is not
 - static : problem of memory wastage may ou occure. Man
 - Dynamic: There no problem of wastage of of memory as memory is allocated at run time as per requirement.

 - · static memory is allocated inside Heap.

 o pynamic memory is allocated inside stack.

of pynamic memory allocation over static memory allocations

-- Advantage: pynamic memory allocation

resolve problem of memory

wastage as memory allocation is done

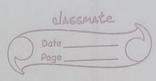
at Run Time.

No memory shortage.

In static memory allocation the calcutions of memory done at compile time a memory allocated at Run time therefore memory wastage & shortage problem occures.

approach same time memory using dynamic approach same time memory is not given to program if there is no space on RAM.

- But in case of static the compilation error will occure in such case.



Q.3 which functions are used in a for dynamic memory allocation 4 deallocation. o To allocate the memory there are three functions in (: (py famic) 1. malloc () 2. calloc c) \ (3518 this sallon & blow 3. realloc () To deallocate the dynamic memory: 1. free() int a pet a work i it was a pointed what functions / operators are used in 9.4 c++ programming for dynamic memory allocation & deallocation. a To allocate memory in c++ programming new () function is used. To deallocate: delete () function is used. for e.g - r classname [+] pointer = new classname delete (): 11 Free memory

what is difference between malloc a 9.5 calloc function.

o malloc function is accepting only one parameter i.e no. of bytes that we want to allocate.

void * malloc (int size) 11 prototype of

· The return value of malloc is void * [pointer which indicates address of allocated memory.

for eq. -> isize = 0; Il user input int * ptr = NULL; 11 NULL pointer

ptr = (int*) malloc (isize * size of (int);

Type casting required as we get address as return : (int*).

Callo C function accepts two parameters. - The first parameter is No. of element A second parameter is size of each element.

void * calloc (int no. of elements.

int size-of-each-element);

- Returns address

- Generally mailor used for Array. for eq. -7

int isize = 0: 11 input from user is

stored. int * Ptr = NULL 11 pointer stored. Pt8 = (int *) calloc (isize, sizeof - Typecast (int +) (int); to of datatupe addresses which is diff Prototype of malloc Function

o prototype void * Malloc (size); - The return type of mailor () is a address. (void *) as (size) no of bytes to be allocated. for eq. ->

int isize = 0; // int * ptr = NULL; 11 ptr = (int*) malloc (isize *(int));

- 4.7 why return value of mailor, callor, reallor function is void.
- the memory address to store that

 we uses pointer (*).
 - And we need different type of

 of pointers to store different types

 of datatype addresses which is dynamically

 assign, 30 for that we need need

 generic pointer void * therefore return

 value of all above functions is void.
- 9.8 what are different uses of realloc function
- -7 · Realloc function is used to resize

 the allocated memory size.
 - o Realloc function used to increase or decrease the size of already allocated memory.
 - o realloc used for memory allocation as well as deallocation.
 - o realloc:

1/ protope void * realloc (void * pt *, int

new size)

memory allocation using malloc

(alloc: int * ptr = (mt *) malloc (6 * size of (in+))

Il Memory allocated 24 bytes

Increasing using realloc

p+r = cint*) realloc(p+r.32)

11 8 bytes increase.

Decreasing using realloc

p+r = cint*) realloc(p+r.16)

11 8 bytes decrease.

9.9 what happens if first parameter of realloc function is NULL9

o If first parameter of realloc ()

is NULL, Then we can used it

is as malloc function

-> void * reallo (NULL * p+8, int new-size)

becomes malloc function

// void * reallo (int new-size);



		1
920	what happens if second parameter of	!
	realloc function is 0)	
	The state of the s	
- Carlotte >	o If we give second parameter	of
-	realloc function o it working	
-	as free function a deallocate	
	the memory	
The sade 2 outs	-> void * realloc (int * ptr. 0)	
- None	11 S phies was one offer of	
-		
		^
- 4		6.0
	realloc function is MULL?	
_	Vol. 1	
	o if first pagameter of realloc o	
	is NULL then we an each it	
	Controlled workers on St.	
(settemen	void & searle (NOLLEPTS , Int	
	becomes mallac Runchion	
	1) yord x yealle t into new si	/
		/
		/
		/
		/
		/
		//