Pointers

Realloc() function

realloc()

- If the dynamically allocated memory is insufficient or more than required, you can change the size of previously allocated memory using the realloc() function.

- Syntax of realloc()

  ptr = realloc(ptr, x);

- Here, ptr is reallocated with a new size x

Example: realloc()

```c
#include <stdio.h>
#include <stdlib.h>
int main() {
    int *ptr, i , n1, n2;
    printf("Enter size: ");
    scanf("%d", &n1);
    ptr = (int*) malloc(n1 * sizeof(int));
    printf("Addresses of previously allocated memory: ");
    for(i = 0; i < n1; ++i)
        printf("%u\n",ptr + i);
    printf("\nEnter the new size: ");
    scanf("%d", &n2);
 // rellocating the memory
    ptr = realloc(ptr, n2 * sizeof(int));
    printf("Addresses of newly allocated memory: ");
    for(i = 0; i < n2; ++i)
        printf("%u\n", ptr + i);
    free(ptr);
    return 0;
}
```

Output:

```
Enter size: 2
Addresses of previously allocated
memory:26855472
26855476
Enter the new size: 4
Addresses of newly allocated memory:26855472
26855476
26855480
26855484
```

Homework

- Write a C program to create an array by dynamically allocating memory using calloc() for 5 integers. Populate the array and print the sum of elements in the array. Use realloc() to resize the memory block from 5 to 8 and populate it. Calculate the new sum.
- Write a C program to create two strings using malloc().Print the two strings. Use realloc() to resize the first string to make it big enough to hold the concatenation of string1 and string2. Print the concatenated string1.