Week 5 – Assignments

1. Write a C program to Allocate Memory Dynamically

```
// Program to calculate the sum of n numbers entered by the user

#include <stdio.h>
#include <stdlib.h>

int main() {
  int n, i, *ptr, sum = 0;

  printf("Enter number of elements: ");
  scanf("%d", &n);

  ptr = (int*) malloc(n * sizeof(int));

  // if memory cannot be allocated
  if(ptr == NULL) {
    printf("Error! memory not allocated.");
    exit(0);
  }

  printf("Enter elements: ");
  for(i = 0; i < n; ++i) {
    scanf("%d", ptr + i);
    sum += *(ptr + i);
  }


  printf("Sum = %d", sum);


  // deallocating the memory

  free(ptr);


  return 0;

}
```

2. Write a C program to illustrate Resizing and Releasing Memory.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main() {

  char name[100];
  char *description;

  strcpy(name, "Zara Ali");

  /* allocate memory dynamically */
  description = malloc( 200 * sizeof(char) );

  if( description == NULL ) {
    fprintf(stderr, "Error - unable to allocate required memory\n");
  } else {
    strcpy( description, "Zara ali a DPS student in class 10th");
  }

  printf("Name = %s\n", name );
  printf("Description: %s\n", description );
}
```

3. Write a C program for deallocation of memory

```
/* code with memory leak */

#include <stdio.h>

#include <stdlib.h>


int main(void)

{

        int *ptr = (int*)malloc(10);


        return 0;

}
```

4. Write a C program for File Handling mechanism

```
#include<stdio.h>
void main( )
{
FILE *fp ;
char ch ;
fp = fopen("file_handle.c","r") ;
while ( 1 )
{
ch = fgetc ( fp ) ;
if ( ch == EOF )
break ;
printf("%c",ch) ;
}
fclose (fp ) ;
}
```

5. Explain in detail about memory leak with appropriate c program

```
/*
 * File: hello.c
 */
#include <stdlib.h>
#include <string.h>

int main(int argc, char *argv[]) {
    char *string, *string_so_far;
    int i, length;     length = 0;
    for(i=0; i<argc; i++) {
        length += strlen(argv[i])+1;
        string = malloc(length+1);

        /*  * Copy the string built so far. */
        if(string_so_far != (char *)0)
            strcpy(string, string_so_far);
        else *string = '\0';
        strcat(string, argv[i]);
        if(i < argc-1) strcat(string, " ");
        string_so_far = string;
    }
    printf("You entered: %s\n", string_so_far);
    return (0);
}
```