**Assigment:5 DATE: 16/01/2024**

**Pratical :1**

**AIM:** Assign Value to the particular place using malloc.

**Code:**

#include <stdlib.h>

int main(){

int \*ptr;

ptr = malloc(15 \* sizeof(\*ptr)); /\* a block of 15 integers \*/

if (ptr != NULL) {

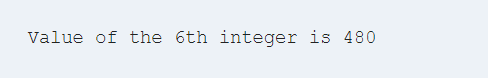
\*(ptr + 5) = 480; /\* assign 480 to sixth integer \*/

printf("Value of the 6th integer is %d",\*(ptr + 5));

}

}

**Output:**



**Pratical :2**

**Aim:** Calculates the sum of an arithmetic sequence using calloc.

**Code:**

#include <stdio.h>

int main() {

int i, \* ptr, sum = 0;

ptr = calloc(10, sizeof(int));

if (ptr == NULL) {

printf("Error! memory not allocated.");

exit(0);

}

printf("Building and calculating the sequence sum of the first 10 terms \ n ");

for (i = 0; i < 10; ++i) { \* (ptr + i) = i;

sum += \* (ptr + i);

}

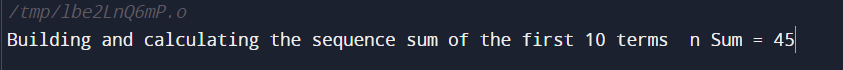
printf("Sum = %d", sum);

free(ptr);

return 0;

}

**Output:**



**Pratical :3**

**Aim:**Write a C program to resize the memory block.

#include <stdio.h>

#include <stdlib.h>

int main() {

int \*ptr = (int\*) malloc(3 \* sizeof(int));

ptr[0] = 1;

ptr[1] = 2;

ptr[2] = 3;

// resize the memory block to hold 5 integers

ptr = (int\*) realloc(ptr, 5 \* sizeof(int));

ptr[3] = 4;

ptr[4] = 5;

for (int i = 0; i< 5; i++) {

printf("%d ", ptr[i]);

}

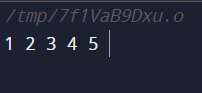
// free the memory block

free(ptr);

return 0;

}

**Output:**



**File Management:**

**Pratical :4**

**Aim:**C Program to illustrate file opening

**Code:**

#include <stdio.h>

#include <stdlib.h>

int main()

{

// file pointer variable to store the value returned by

// fopen

FILE\* fptr;

// opening the file in read mode

fptr = fopen("C:\filename.txt", "r");

// checking if the file is opened successfully

if (fptr == NULL) {

printf("The file is not opened. The program will "

"now exit.");

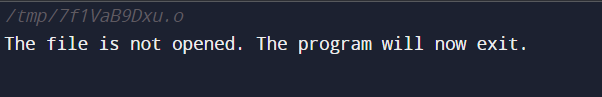
exit(0);

}

return 0;

}

**Output:**



**Pratical :5**

**Aim:** C Program to create a file