

## Wap to find the sum of elements using pointers

~

```
#include <stdio.h>

int main() {
    int arr[] = {1,2,3,4,5}, sum=0;
    for (int i = 0; i < sizeof(arr)/sizeof(arr[0]); i++)    sum += *(arr+i) ;
    printf("%d", sum);
}
```

~

```
[wizard@archlinux labreport]$ ./a.out
15
[wizard@archlinux labreport]$
```

## Wap to search for an element using pointers.

~

```
#include <stdio.h>

int main() {
    int arr[] = {1,2,3,4,5}, toFind=5;
    for (int i = 0; i < sizeof(arr)/sizeof(arr[0]); i++) {
        if (toFind == *(arr+i)) {
            printf("Found at index %d", i);
            return 0;
        }
    }
    printf("Couldnt find the thing");
}
```

~

```
[wizard@archlinux labreport]$ ./a.out
Found at index 4
[wizard@archlinux labreport]$
```

## WAP that allocates memory for 3 arrays, the user inputs the values into 2 of them, which is stored into the third.

~

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

int main(){
    int n;
    scanf("%d",&n);
    int *arr1 = malloc(n * sizeof(int)), *arr2 = malloc(n * sizeof(int)), *arr3 = malloc(n * sizeof(int));
    for (int i = 0; i < n; i++) scanf("%d",&arr1[i]);
    for (int i = 0; i < n; i++) scanf("%d",&arr2[i]);
    for(int i = 0; i<n ;i++){
        printf("%d, ", arr1[i] + arr2[i]);
        arr3[i] = arr1[i] + arr2[i];
    }
    free(arr1);free(arr2);free(arr3);
}
```

~

```
[wizard@archlinux labreport]$ ./a.out
3
1
2
3
4
5
6
5, 7, 9
[wizard@archlinux labreport]$ 1
```

## WAP to find the max in an array using dynamic array allocation.

~

```
#include <stdlib.h>

int main(){
    int n,max = 0;
    scanf("%d",&n);
    int *arr = malloc(n * sizeof(int));
    for (int i = 0; i < n; i++) {
        scanf("%d",&arr[i]);
        max = (arr[i] > max)? arr[i]:max;
    }
    printf("Max: %d",max);
}
```

~

```
4
1
2
6
9
Max: 9[wizard@archlinux labreport]$
```

## Wap to create memory for char, int and float at runtime

~

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

int main(){
    int *n = malloc(1 * sizeof(int));
    char *a = malloc(1);
    float *b = malloc(1 * sizeof(float));
    scanf("%d %c %f",n,a,b);
    printf("Got: %d, %c, %f",*n,*a,*b);
    free(n);free(a);free(b);
}
```

~

```
[wizard@archlinux labreport]$ ./a.out
1 a 2.0
Got: 1, a, 2.000000
[wizard@archlinux labreport]$
```

## WAP to take an array, using dynmaic memory allocation and print them, later add elements into that array using realloc and print them all

~

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

int main(){
    int n=3,nn=5;
    int *arr = malloc(n * sizeof(int));
    for (int i = 0; i < n; i++) scanf("%d", (arr+i));
    for (int i = 0; i < n; i++) printf("%d",*(arr+i));
    arr = realloc(arr,nn);
    for (int i = n; i < nn; i++) scanf("%d", (arr+i));
    for (int i = 0; i < nn; i++) printf("%d",*(arr+i));
    free(arr);
}
```

~

```
[wizard@archlinux labreport]$ gcc labreport.c
[wizard@archlinux labreport]$ ./a.out
1
2
3
123
4
5
12345
[wizard@archlinux labreport]$
```