Universidad Nacional Autónoma De México

Profesor: *Marco Antonio Martínez*

Asignatura*: Estructura de datos y algoritmos 1*

Grupo: *17*

No de Práctica(s): *3 Tipo de dato abstracto*

Alumno: *Carvajal Axol Brandon Emir*

Semestre: *2020-2*

Observaciones:

CALIFICACIÓN: _____

Malloc
(Muestra datos basura en la memoria o residuos de programas anteriores)

```c
#include <stdio.h>
#include <stdlib.h>
int main (){
int *arreglo, num, cont;
//Te pide valor del conjunto
printf("¿Cuantos elementos tiene el conjunto?\n");
scanf("%d",&num);
//Se usa "Malloc"
//Reserva espacio de memoria
arreglo = (int *)malloc (num * sizeof(int));
//Imprime los valores de datos guardagos en nuestro conjunto
if (arreglo!=NULL) {
printf("Vector reservado:\n\t[");
for (cont=0 ; cont<num ; cont++){
printf("\t%d",*(arreglo+cont));
}
printf("\t]\n");
printf("Se libera el espacio reservado.\n");
free(arreglo);
}
return 0;
}
```

Te pide cuantos elementos tiene el conjunto y la función malloc te muestra el valor de los dados que están guardados en ese espacio.

```
¿Cuantos elementos tiene el conjunto?
2
Vector reservado:
        [       12661304        12648640        ]
Se libera el espacio reservado.

Process returned 0 (0x0)   execution time : 3.318 s
Press any key to continue.
```

```
¿Cuantos elementos tiene el conjunto?
3
Vector reservado:
        [       7405760 7414288 0       ]
Se libera el espacio reservado.

Process returned 0 (0x0)   execution time : 1.992 s
Press any key to continue.
```

```
¿Cuantos elementos tiene el conjunto?
7
Vector reservado:
        [       131264  139792  0       0       0       0       0       ]
Se libera el espacio reservado.

Process returned 0 (0x0)   execution time : 2.025 s
Press any key to continue.
```

Calloc
(Limpia memoria)

```c
#include <stdio.h>
#include <stdlib.h>
int main (){
int *arreglo, num, cont;
//Te pide valor del conjunto
printf("¿Cuantos elementos tiene el conjunto?\n");
scanf("%d",&num);
//Se usa "Calloc"
//Reserva espacio de memoria y los deja en valor cero
arreglo = (int *)calloc (num, sizeof(int));
//Imprime el vector de memoria limpia
if (arreglo!=NULL) {
printf("Vector reservado:\n\t[");
for (cont=0 ; cont<num ; cont++){
printf("\t%d",*(arreglo+cont));
}
printf("\t]\n");
printf("Se libera el espacio reservado.\n");
free(arreglo);
}
return 0;
}
```

La función calloc funciona de manera similar a la función malloc pero, además de reservar memoria en tiempo real, inicializa la memoria reservada con 0.

```
¡Cuantos elementos tiene el conjunto?
2
Vector reservado:
        [       0       0       ]
Se libera el espacio reservado.

Process returned 0 (0x0)   execution time : 4.021 s
Press any key to continue.
```

```
¡Cuantos elementos tiene el conjunto?
3
Vector reservado:
        [       0       0       0       ]
Se libera el espacio reservado.

Process returned 0 (0x0)   execution time : 1.887 s
Press any key to continue.
```

```
¡Cuantos elementos tiene el conjunto?
7
Vector reservado:
        [       0       0       0       0       0       0       0       ]
Se libera el espacio reservado.

Process returned 0 (0x0)   execution time : 1.835 s
Press any key to continue.
```

Realloc
(Redimensiona el espacio de memoria)

```c
#include <stdio.h>
#include <stdlib.h>
int main (){
int *arreglo, *arreglo2, num, cont;
//Te pide valor del conjunto
printf("¿Cuántos elementos tiene el conjunto?\n");
scanf("%d",&num);
//Se usa "Malloc"
arreglo = (int *)malloc (num * sizeof(int));
if (arreglo!=NULL) {
for (cont=0 ; cont < num ; cont++){
printf("Inserte el elemento %d del conjunto.\n",cont+1);
scanf("%d",(arreglo+cont));
}
printf("Vector insertado:\n\t[");
for (cont=0 ; cont < num ; cont++){
printf("\t%d",*(arreglo+cont));
}
printf("\t]\n");
printf("Aumentando el tamaño del conjunto al doble.\n");
num *= 2;
//Se usa la funcion realloc
//Aumenta el tamaño de memoria reservada
arreglo2 = (int *)realloc (arreglo,num*sizeof(int));
if (arreglo2 != NULL) {
arreglo = arreglo2;
//Ingresas nuevos valors a nuestra nueva dimencion de memoria
for (; cont < num ; cont++){
printf("Inserte el elemento %d del conjunto.\n",cont+1);
scanf("%d",(arreglo2+cont));
}
printf("Vector insertado:\n\t[");
for (cont=0 ; cont < num ; cont++){
printf("\t%d",*(arreglo2+cont));
}
printf("\t]\n");
}
free (arreglo);
}
return 0;
}
```

La función realloc permite redimensionar el espacio asignado previamente de forma dinámica, es decir, permite aumentar el tamaño de la memoria reservada de manera dinámica. Su sintaxis es la siguiente:

Si el apuntador que se desea redimensionar tiene el valor nulo, la función actúa como la función malloc. Si la reasignación no se pudo realizar, la función devuelve un apuntador a nulo, dejando intacto el apuntador que se pasa como parámetro (el espacio reservado previamente).