

1- Write a program that dynamically allocates memory for a character string using malloc and prints it.

اكتب برنامجًا يخصص الذاكرة ديناميكيًا لسلسلة أحرف باستخدام malloc ويطبعها.

Output

```
String: Hello
```

Solution

```
// www.gammal.tech

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

int main() {
    // Allocate memory for a character string
    char *str = (char *)malloc(10 * sizeof(char));

    // Assign a value to the allocated memory
    strcpy(str, "Hello");

    // Print the string
    printf("String: %s\n", str);

    // Free the allocated memory
    free(str);

    return 0;
}
```

2- Write a program that dynamically allocates memory for an integer array of size 3 using malloc and initializes it with values 2, 4, and 6. Then, use realloc to resize the array to size 5 and assign values 8 and 10 to the new elements.

اكتب برنامجًا يخصص الذاكرة ديناميكيًا array أعداد صحيحة بالحجم 3 باستخدام malloc وتهيئتها بالقيم 2 و 4 و 6. ثم استخدم realloc لتغيير حجم realloc إلى الحجم 5 وتعيين القيمتين 8 و 10 للعناصر الجديدة.

Output

```
Initial Array: 2 4 6
Final Array: 2 4 6 8 10
```

Solution

```
// www.gammal.tech

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

int main() {
    // Allocate memory for an integer array of size 3
    int *arr = (int *)malloc(3 * sizeof(int));

    // Initialize the array with values 2, 4, and 6
    for (int i = 0; i < 3; i++)
        arr[i] = (i + 1) * 2;

    // Print the initial array elements
    printf("Initial Array: ");
    for (int i = 0; i < 3; i++)
        printf("%d ", arr[i]);
    printf("\n");

    // Resize the array to size 5 using realloc
    arr = (int *)realloc(arr, 5 * sizeof(int));

    // Assign values 8 and 10 to the new elements
    arr[3] = 8;
    arr[4] = 10;

    // Print the final array elements
    printf("Final Array: ");
    for (int i = 0; i < 5; i++)
        printf("%d ", arr[i]);
    printf("\n");

    // Free the allocated memory
    free(arr);

    return 0;
}
```

3- Write a program that dynamically allocates memory for a floating-point array of size 2 using malloc and initializes it. Then, use realloc to resize the array to size 5 and assign new values.

اكتب برنامجًا يخصص الذاكرة ديناميكيًا array فاصلة floating بحجم 2 باستخدام malloc وتهيئتها. ثم استخدم realloc لتغيير حجم array إلى الحجم 5 وتعيين قيم جديدة.

Output

```
Initial Array: 2.50 3.50
Final Array: 2.50 3.50 4.50 5.50 6.50
```

Solution

```
// www.gammal.tech

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

int main() {
    // Allocate memory for a floating-point array of size 2
    double *arr = (double *)malloc(2 * sizeof(double));

    // Initialize the array with values 2.5 and 3.5
    arr[0] = 2.5;
    arr[1] = 3.5;

    // Print the initial array elements
    printf("Initial Array: %.2lf %.2lf\n", arr[0], arr[1]);

    // Resize the array to size 5 using realloc
    arr = (double *)realloc(arr, 5 * sizeof(double));

    // Assign new values to the additional elements
    arr[2] = 4.5;
    arr[3] = 5.5;
    arr[4] = 6.5;

    // Print the final array elements
    printf("Final Array: %.2lf %.2lf %.2lf %.2lf %.2lf\n", arr[0], arr[1], arr[2], arr[3], arr[4]);

    // Free the allocated memory
    free(arr);

    return 0;
}
```

4- Write a program that dynamically allocates memory for an array of integers using malloc, initializes it with values 1 to 3, and then uses realloc to increase the size of the array to 5, assigning new values 4 and 5.

اكتب برنامجًا يخصص الذاكرة ديناميكيًا array من الأعداد الصحيحة باستخدام malloc، وتهيئتها بالقيم من 1 إلى 3، ثم يستخدم realloc لزيادة حجم array إلى 5، وتعيين قيمتين جديدتين 4 و5.

Output

```
Initial Array: 1 2 3
Final Array: 1 2 3 4 5
```

Solution

```
// www.gammal.tech

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

int main() {
    // Allocate memory for an integer array of size 3
    int *arr = (int *)malloc(3 * sizeof(int));

    // Initialize the array with values 1, 2, and 3
    for (int i = 0; i < 3; i++)
        arr[i] = i + 1;

    // Print the initial array elements
    printf("Initial Array: %d %d %d\n", arr[0], arr[1], arr[2]);

    // Resize the array to size 5 using realloc
    arr = (int *)realloc(arr, 5 * sizeof(int));

    // Assign new values to the additional elements
    arr[3] = 4;
    arr[4] = 5;

    // Print the final array elements
    printf("Final Array: %d %d %d %d %d\n", arr[0], arr[1], arr[2], arr[3], arr[4]);

    // Free the allocated memory
    free(arr);

    return 0;
}
```

5- Write a program that dynamically allocates memory for an array of integers using malloc, initializes it with values 10, 20, and 30, and then uses realloc to resize the array to size 2.

اكتب برنامجًا يخصص الذاكرة ديناميكيًا لمصفوفة من الأعداد الصحيحة باستخدام malloc، وتهيئتها بالقيم 10 و 20 و 30، ثم يستخدم realloc لتغيير حجم array إلى الحجم 2.

Output

```
Initial Array: 10 20 30
Final Array: 10 20
```

Solution

```
// www.gammal.tech

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

int main() {
    // Allocate memory for an integer array of size 3
    int *arr = (int *)malloc(3 * sizeof(int));

    // Initialize the array with values 10, 20, and 30
    for (int i = 0; i < 3; i++)
        arr[i] = (i + 1) * 10;

    // Print the initial array elements
    printf("Initial Array: %d %d %d\n", arr[0], arr[1], arr[2]);

    // Resize the array to size 2 using realloc
    arr = (int *)realloc(arr, 2 * sizeof(int));

    // Print the final array elements
    printf("Final Array: %d %d\n", arr[0], arr[1]);

    // Free the allocated memory
    free(arr);

    return 0;
}
```

6- Write a program to dynamically allocate memory for an array of 3 integers using calloc and initialize them with values 5, 10, and 15. Then use realloc to resize the array to hold 5 integers and initialize the new elements with values 20 and 25.

اكتب برنامجًا لتخصيص الذاكرة ديناميكيًا array مكونة من 3 أعداد صحيحة باستخدام calloc وتهيئتها بالقيم 5 و10 و15. ثم استخدم إعادة التخصيص لتغيير حجم array لاستيعاب 5 أعداد صحيحة وتهيئة العناصر الجديدة بالقيم 20 و25.

Output

```
Memory before realloc: 0xae92a0
Memory after realloc: 0xae92a0
5 10 15 20 25
```

Solution

```
// www.gammal.tech

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

int main() {
    int *x = (int *)calloc(3, sizeof(int));
    printf("Memory before realloc: %p\n", (void*)x);

    for (int i = 0; i < 3; i++)
        x[i] = (i + 1) * 5;

    x = (int *)realloc(x, 5 * sizeof(int));
    printf("Memory after realloc: %p\n", (void*)x);

    for (int i = 3; i < 5; i++)
        x[i] = (i + 1) * 5;

    for (int i = 0; i < 5; i++)
        printf("%d ", x[i]);

    free(x);
    return 0;
}
```

7- Write a program to dynamically allocate memory for an array of 4 characters using calloc and initialize them with characters 'A', 'B', 'C', and 'D'. Then use realloc to resize the array to hold 6 characters and initialize the new elements with characters 'E' and 'F'.

اكتب برنامجًا لتخصيص الذاكرة ديناميكيًا لمجموعة مكونة من 4 أحرف باستخدام `calloc` وتهيئتها بالأحرف "A" و "B" و "C" و "D". ثم استخدم `realloc` لتغيير حجم `array` لاستيعاب 6 أحرف وتهيئة العناصر الجديدة بالأحرف "E" و "F".

Output

```
Memory before realloc: 0x19942a0
Memory after realloc: 0x19942a0
A B C D E F
```

Solution

```
// www.gammal.tech

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

int main() {
    char *x = (char *)calloc(4, sizeof(char));
    printf("Memory before realloc: %p\n", (void*)x);

    for (int i = 0; i < 4; i++)
        x[i] = 'A' + i;

    x = (char *)realloc(x, 6 * sizeof(char));
    printf("Memory after realloc: %p\n", (void*)x);

    for (int i = 4; i < 6; i++)
        x[i] = 'E' + (i - 4);

    for (int i = 0; i < 6; i++)
        printf("%c ", x[i]);

    free(x);
    return 0;
}
```

8- Write a program to dynamically allocate memory for an array of 5 doubles using `calloc` and initialize them with values 2.5, 5.0, 7.5, 10.0, and 12.5. Then use `realloc` to resize the array to hold 7 doubles and initialize the new elements with values 15.0 and 17.5.

اكتب برنامجًا لتخصيص الذاكرة ديناميكيًا array مكونة من 5 double باستخدام calloc وقم بتهيئتها بالقيم 2.5، 5.0، 7.5، 10.0، و12.5. ثم استخدم إعادة التخصيص لتغيير حجم array لاستيعاب 7 مضاعفات وتهيئة العناصر الجديدة بالقيمتين 15.0 و17.5.

Output

```
Memory before realloc: 0x4242a0
Memory after realloc: 0x4246e0
2.5 5.0 7.5 10.0 12.5 15.0 17.5
```

Solution

```
// www.gammal.tech

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

int main() {
    double *x = (double *)calloc(5, sizeof(double));
    printf("Memory before realloc: %p\n", (void*)x);

    for (int i = 0; i < 5; i++)
        x[i] = (i + 1) * 2.5;

    x = (double *)realloc(x, 7 * sizeof(double));
    printf("Memory after realloc: %p\n", (void*)x);

    for (int i = 5; i < 7; i++)
        x[i] = (i + 1) * 2.5;

    for (int i = 0; i < 7; i++)
        printf("%.1lf ", x[i]);

    free(x);
    return 0;
}
```

9- Write a program to dynamically allocate memory for an array of 3 integers using calloc and initialize them with values 3, 6, and 9. Then use realloc to resize the array to hold 4 integers and initialize the new element with the value 12.

اكتب برنامجًا لتخصيص الذاكرة ديناميكيًا array مكونة من 3 أعداد صحيحة باستخدام calloc وتهيئتها بالقيم 3 و6 و9. ثم استخدم realloc لتغيير حجم array لاستيعاب 4 أعداد صحيحة وتهيئة العنصر الجديد بالقيمة 12.

Output

```
Memory before realloc: 0x4d82a0
Memory after realloc: 0x4d82a0
3 6 9 12
```

Solution

```
// www.gammal.tech

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

int main() {
    int *x = (int *)calloc(3, sizeof(int));
    printf("Memory before realloc: %p\n", (void*)x);

    for (int i = 0; i < 3; i++)
        x[i] = (i + 1) * 3;

    x = (int *)realloc(x, 4 * sizeof(int));
    printf("Memory after realloc: %p\n", (void*)x);

    x[3] = 12;

    for (int i = 0; i < 4; i++)
        printf("%d ", x[i]);

    free(x);
    return 0;
}
```

10- Write a program to dynamically allocate memory for an array of 6 doubles using calloc and initialize them with values 2.5, 5.0, 7.5, 10.0, 12.5, and 15.0. Then use realloc to resize the array to hold 4 doubles and initialize the new elements with values 17.5, 20.0, 22.5, and 25.0.

اكتب برنامجًا لتخصيص الذاكرة ديناميكيًا array مكونة من 6 وحدات double باستخدام calloc وقم بتهيئتها بالقيم 2.5، 5.0، 7.5، 10.0، 12.5، و15.0. ثم استخدم realloc لتغيير حجم array لاستيعاب 4 مضاعفات وتهيئة العناصر الجديدة بالقيم 17.5 و20.0 و22.5 و25.0.

Output

```
Memory before realloc: 0x47d2a0
Memory after realloc: 0x47d2a0
2.5 5.0 7.5 10.0 12.5 15.0 17.5 20.0 0.0 0.0
```

Solution

```
// www.gammal.tech

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

int main() {
    int *x = (int *)calloc(3, sizeof(int));
    printf("Memory before realloc: %p\n", (void*)x);

    for (int i = 0; i < 3; i++)
        x[i] = (i + 1) * 3;

    x = (int *)realloc(x, 4 * sizeof(int));
    printf("Memory after realloc: %p\n", (void*)x);

    x[3] = 12;

    for (int i = 0; i < 4; i++)
        printf("%d ", x[i]);

    free(x);
    return 0;
}

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

int main() {
    double *x = (double *)calloc(6, sizeof(double));
    printf("Memory before realloc: %p\n", (void*)x);

    for (int i = 0; i < 6; i++)
        x[i] = (i + 1) * 2.5;

    x = (double *)realloc(x, 4 * sizeof(double));
    printf("Memory after realloc: %p\n", (void*)x);

    for (int i = 6; i < 10; i++)
        x[i] = (i + 1) * 2.5;

    for (int i = 0; i < 10; i++)
        printf("%.1lf ", x[i]);

    free(x);
    return 0;
}
```