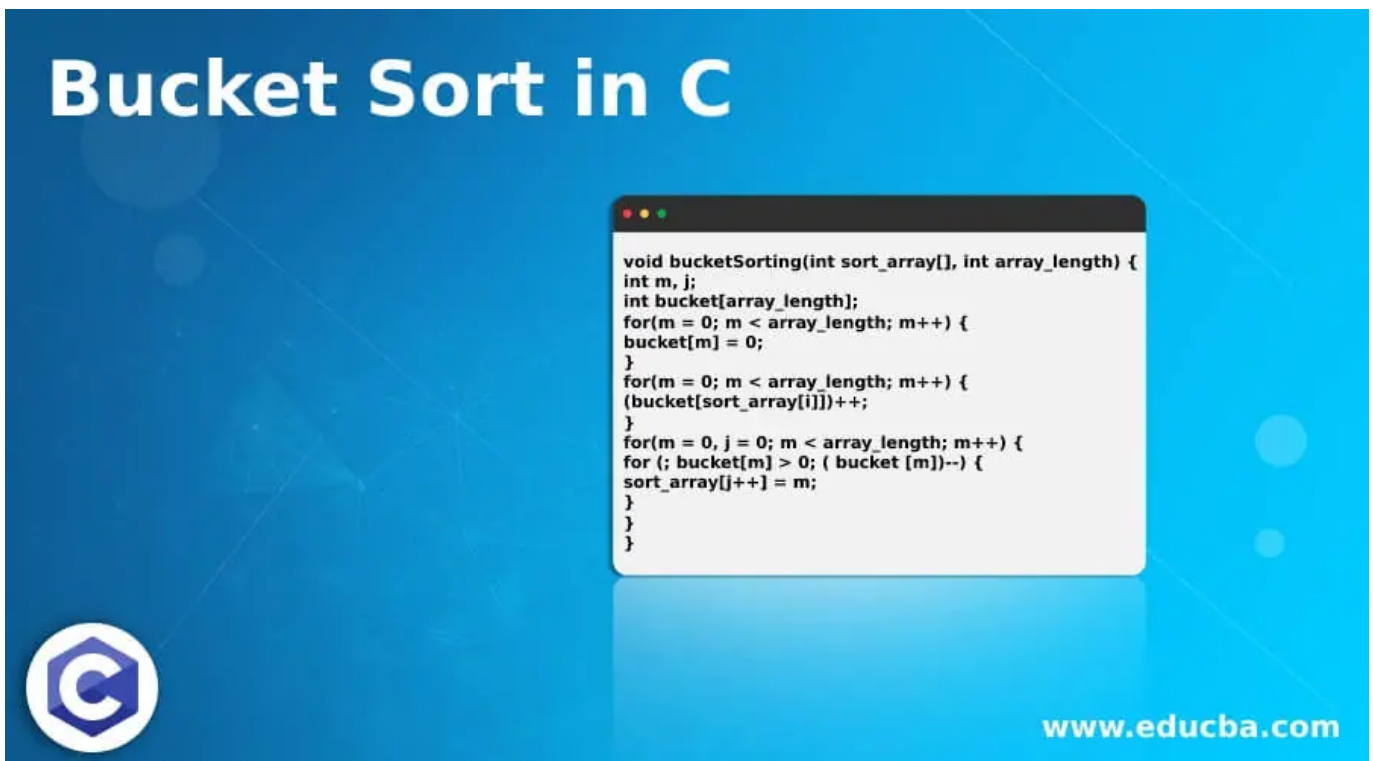




(<https://www.educba.com/software-development/>)

← (<https://www.educba.com/decimal-to-hexadecimal-in-c/>)



Introduction to Bucket Sort in C

The bucket sort is a sorting algorithm to arrange elements in ascending order using a C programming language. It helps to partition array elements in a respective bucket as per element range using C language. The bucket sort is a sorting algorithm to arrange element

range and keep it as per ascending or descending order. The bucket sort is a sorting algorithm to combine methods set up, scatter, sorting, and gather the array element. It is creating buckets as per element range and sorting array elements in ascending order using a C programming





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

you need an algorithm to know about bucket sort syntax.

Start Your Free Software Development Course

Web development, programming languages, Software testing & others

The bucket sort algorithm is below:

- Create an empty bucket to place the array element.
- Insert array element in the bucket as per element range.
- Classify array elements as per required order.
- Write down the array element as per the sequence.

Bucket sort syntax is below:

```
void bucketSorting(int sort_array[], int array_length) {  
    int m, j;  
    int bucket[array_length];  
    for(m = 0; m < array_length; m++) {  
        bucket[m] = 0;  
    }  
    for(m = 0; m < array_length; m++) {  
        (bucket[sort_array[m]])++;  
    }  
    for(m = 0, j = 0; m < array_length; m++) {  
        for (; bucket[m] > 0; (bucket[m])--) {  
            sort_array[j++] = m;  
        }  
    }  
}
```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

HOW DOES BUCKET SORT WORK IN C?

Given below is the working:

1. Create an empty bucket to place the array element.

```
struct buckets
{
    int number;
    int* element;
};
```

2. Compare array elements in the bucket as per element range.

```
int comparevalues(const void* fvalue, const void* svalue)
{
    int p = *((int*)fvalue), q = *((int*)svalue);
    if (p == q)
    {
        return 0;
    }
    else if (p < q)
    {
        return -1;
    }
    else
```





<https://www.educba.com/software-development/>

Popular Course in this category



C Programming Training (3 Courses, 5 Project)

3 Online Courses | 5 Hands-on Projects | 34+ Hours | Verifiable Certificate of Completion | Lifetime Access

★★★★★ 4.5 (8,644 ratings)

Course Price

\$79 ~~\$399~~

[View Course](#)

(<https://www.educba.com/software-development/courses/c-programming-course/?btnz=edu-blg-inline-banner1>)

Related Courses

C++ Training (4 Courses, 5 Projects, 4 Quizzes) (<https://www.educba.com/software-development/courses/c-course/?btnz=edu-blg-inline-banner1>)

Java Training (40 Courses, 29 Projects, 4 Quizzes) (<https://www.educba.com/software-development/courses/java-course/?btnz=edu-blg-inline-banner1>)

3. Classify array element as per required order.

```
void crtbucketsortingElement(int sortarry[], int arylength)
{
    struct buckets crtbuckets[3];
```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
crtbuckets[m].element = (int*)malloc(sizeof(int) * arylength);
}
for (m = 0; m < arylength; m++)
{
    if (sortarry[m] < 0)
    {
        crtbuckets[0].element[crtbuckets[0].number++] = sortarry[m];
    }
    else if (sortarry[m] > 10)
    {
        crtbuckets[2].element[crtbuckets[2].number++] = sortarry[m];
    }
    else
    {
        crtbuckets[1].element[crtbuckets[1].number++] = sortarry[m];
    }
}
for (k = 0, m = 0; m < 3; m++)
{
    qsort(crtbuckets[m].element, crtbuckets[m].number, sizeof(int),
    &comparevalues);
    for (j = 0; j < crtbuckets[m].number; j++)
    {
        sortarry[k + j] = crtbuckets[m].element[j];
    }
}
```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

4. Write down the array element as per the sequence.

```
int main(char *arg[]) {  
    int sortarray[100] = { 5, 9, 1, 2, 3, 2, 5, 4, 7, 0, 8, 6 };  
    int m = 12,j,k,arylength;  
    arylength = m;  
    printf("Before bucket Sorting array element: \n");  
    for (j = 0; j<m; j++)  
    {  
        printf("%d ", sortarray[j]);  
    }  
    crtbucketsortingElement(sortarray, arylength);  
    printf("\n After bucket Sorting array element: \n");  
    for (k = 0; k<m; k++)  
        printf("%d ", sortarray[k]);  
    return 0;  
}
```

Examples of Bucket Sort in C

Given below are the examples mentioned:

Example #1

The bucket sort with positive value example and output are below.

Code:





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
int number;
int* element;
};

int comparevalues(const void* fvalue, const void* svalue)
{
    int p = *((int*)fvalue), q = *((int*)svalue);
    if (p == q)
    {
        return 0;
    }
    else if (p < q)
    {
        return -1;
    }
    else
    {
        return 1;
    }
}

void crtbucketsortingElement(int sortarry[], int arylength)
{
    struct buckets crtbuckets[3];

    int m, j, k;
    for (m = 0; m < 3; m++)
    {
```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```

{
if (sortarry[m] < 0)
{
crtbuckets[0].element[crtbuckets[0].number++] = sortarry[m];
}
else if (sortarry[m] > 10)
{
crtbuckets[2].element[crtbuckets[2].number++] = sortarry[m];
}
else
{
crtbuckets[1].element[crtbuckets[1].number++] = sortarry[m];
}
}
for (k = 0, m = 0; m < 3; m++)
{
qsort(crtbuckets[m].element, crtbuckets[m].number, sizeof(int),
&comparevalues);
for (j = 0; j < crtbuckets[m].number; j++)
{
sortarry[k + j] = crtbuckets[m].element[j];
}
k = k+ crtbuckets[m].number;
free(crtbuckets[m].element);
}
}
```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
arylength = m,
printf("Before bucket Sorting array element: \n");
for (j = 0; j<m; j++)
{
printf("%d ", sortarry[j]);
}
crtbucketsortingElement(sortarry, arylength);
printf("\nAfter bucket Sorting array element: \n");
for (k = 0; k<m; k++)
printf("%d ", sortarry[k]);
return 0;
}
```

Output:

Example #2

The bucket sort with negative value example and output are below.

Code:

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





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
int comparevalues(const void* fvalue, const void* svalue)
{
    int p = *((int*)fvalue), q = *((int*)svalue);
    if (p == q)
    {
        return 0;
    }
    else if (p < q)
    {
        return -1;
    }
    else
    {
        return 1;
    }
}

void crtbucketsortingElement(int sortarry[], int arylength)
{
    struct buckets crtbuckets[3];
    int m, j, k;
    for (m = 0; m < 3; m++)
    {

        crtbuckets[m].number = 0;
        crtbuckets[m].element = (int*)malloc(sizeof(int) * arylength);
    }
}
```





(<https://www.educba.com/software-development/>)

```

crtbuckets[0].element[crtbuckets[0].number++] = sortarry[m],
}
else if (sortarry[m] > 10)
{
crtbuckets[2].element[crtbuckets[2].number++] = sortarry[m];
}
else
{
crtbuckets[1].element[crtbuckets[1].number++] = sortarry[m];
}
}
for (k = 0, m = 0; m < 3; m++)
{
qsort(crtbuckets[m].element, crtbuckets[m].number, sizeof(int),
&comparevalues);
for (j = 0; j < crtbuckets[m].number; j++)
{
sortarry[k + j] = crtbuckets[m].element[j];
}
k = k+ crtbuckets[m].number;
free(crtbuckets[m].element);
}
}

int main(char *arg[]) {
int sortarry[100] = { -5, -91, -1, -112, -3, -2673, -54, -4000,
-781, -100, -8, -61 };

```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
    printf("%d ", sortarry[j]);  
}  
crtbucketsortingElement(sortarry, arylength);  
printf("\nAfter bucket Sorting array element: \n");  
for (k = 0; k<m; k++)  
    printf("%d ", sortarry[k]);  
return 0;  
}
```

Output:

Example #3

The bucket sort with combine values example and output are below.

Code:

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





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
int comparevalues(const void* fvalue, const void* svalue)
{
    int p = *((int*)fvalue), q = *((int*)svalue);
    if (p == q)
    {
        return 0;
    }
    else if (p < q)
    {
        return -1;
    }
    else
    {
        return 1;
    }
}

void crtbucketsortingElement(int sortarry[], int arylength)
{
    struct buckets crtbuckets[3];
    int m, j, k;
    for (m = 0; m < 3; m++)
    {

        crtbuckets[m].number = 0;
        crtbuckets[m].element = (int*)malloc(sizeof(int) * arylength);
    }
}
```





(<https://www.educba.com/software-development/>)

```

crtbuckets[0].element[crtbuckets[0].number++] = sortarry[m];
}
else if (sortarry[m] > 10)
{
crtbuckets[2].element[crtbuckets[2].number++] = sortarry[m];
}
else
{
crtbuckets[1].element[crtbuckets[1].number++] = sortarry[m];
}
}
for (k = 0, m = 0; m < 3; m++)
{
qsort(crtbuckets[m].element, crtbuckets[m].number, sizeof(int),
&comparevalues);
for (j = 0; j < crtbuckets[m].number; j++)
{
sortarry[k + j] = crtbuckets[m].element[j];
}
k = k+ crtbuckets[m].number;
free(crtbuckets[m].element);
}
}

int main(char *arg[]) {
int sortarry[100] = { -5, 91, -1, -112, 3, -2673, -54, 4000, -781,
100, 8, 61 };

```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
    printf("%d ", sortarry[j]);  
}  
crtbucketsortingElement(sortarry, arylength);  
printf("\nAfter bucket Sorting array element: \n");  
for (k = 0; k<m; k++)  
    printf("%d ", sortarry[k]);  
return 0;  
}
```

Output:

Conclusion

The bucket sort helps to arrange elements using an empty bucket. Its language helps to arrange array elements as per the user's requirement. The bucket sort arranges data without the complexity and makes a user-friendly application.

Recommended Articles

This is a guide to Bucket Sort in C. Here we discuss the introduction, working and examples to get a better understanding. You may also have a look at the following articles to learn more –



1. Pointer Arithmetic in C (<https://www.educba.com/pointer-arithmetic-in-c/>)



[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

ALL IN ONE SOFTWARE DEVELOPMENT BUNDLE (600+ COURSES, 50+ PROJECTS)

- ☒ 600+ Online Courses
- ☒ 50+ projects
- ☒ 3000+ Hours
- ☒ Verifiable Certificates
- ☒ Lifetime Access

Learn More

[\(https://www.educba.com/software-development/courses/software-development-course/?btnz=edu-blg-inline-banner3\)](https://www.educba.com/software-development/courses/software-development-course/?btnz=edu-blg-inline-banner3)

About Us

Blog (<https://www.educba.com/blog/?source=footer>)

Who is EDUCBA? (<https://www.educba.com/about-us/?source=footer>)





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

Contact Us (<https://www.educba.com/contact-us/?source=footer>)

Verifiable Certificate (<https://www.educba.com/software-development/verifiable-certificate/?source=footer>)

Reviews (<https://www.educba.com/software-development/reviews/?source=footer>)

Terms and Conditions (<https://www.educba.com/terms-and-conditions/?source=footer>)

Privacy Policy (<https://www.educba.com/privacy-policy/?source=footer>)

Apps

iPhone & iPad (<https://itunes.apple.com/in/app/educba-learning-app/id1341654580?mt=8>)

Android (<https://play.google.com/store/apps/details?id=com.educba.www>)

Resources

Free Courses (<https://www.educba.com/software-development/free-courses/?source=footer>)

Java Tutorials (<https://www.educba.com/software-development/software-development-tutorials/java-tutorial/?source=footer>)

Python Tutorials (<https://www.educba.com/software-development/software-development-tutorials/python-tutorial/?source=footer>)

All Tutorials (<https://www.educba.com/software-development/software-development-tutorials/?source=footer>)

Certification Courses

All Courses (<https://www.educba.com/software-development/courses/?source=footer>)





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

course/?source=footer)

Become a Selenium Automation Tester (<https://www.educba.com/software-development/courses/selenium-training-certification/?source=footer>)

Become an IoT Developer (<https://www.educba.com/software-development/courses/iot-course/?source=footer>)

ASP.NET Course (<https://www.educba.com/software-development/courses/asp-net-course/?source=footer>)

VB.NET Course (<https://www.educba.com/software-development/courses/vb-net-course/?source=footer>)

PHP Course (<https://www.educba.com/software-development/courses/php-course/?source=footer>)

© 2022 - EDUCBA. ALL RIGHTS RESERVED. THE CERTIFICATION NAMES ARE THE TRADEMARKS OF THEIR RESPECTIVE OWNERS.

