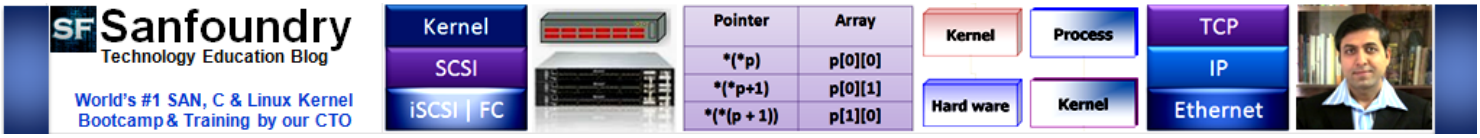


(http://www.sanfoundry.com)

Questions & Answers

C Interview Questions (http://www.sanfoundry.com/c-interview-questions-answers/)
C++ Questions (http://www.sanfoundry.com/cplusplus-interview-questions-answers/)
Linux MCQs (http://www.sanfoundry.com/technical-interview-questions/)
C# Quiz (http://www.sanfoundry.com/csharp-questions-answers/)
Java MCQs (http://www.sanfoundry.com/java-questions-answers-freshers-experienced/)
JavaScript MCQs (http://www.sanfoundry.com/javascript-questions-answers/)
SAN Questions (http://www.sanfoundry.com/san-storage-mcqs-freshers-experienced/)
PHP Questions (http://www.sanfoundry.com/php-questions-answers/)
Python Quiz (http://www.sanfoundry.com/1000-python-questions-answers/)

Computer Science Questions

Operating System Quiz (http://www.sanfoundry.com/operating-system-questions-answers/)
Computer Architecture MCQs (http://www.sanfoundry.com/com

puter-organization-architecture-questions-answers/)
Software Architecture MCQs (http://www.sanfoundry.com/software-architecture-design-questions-answers/)
Software Engineering MCQs (http://www.sanfoundry.com/software-engineering-questions-answers/)
Artificial Intelligence MCQs (http://www.sanfoundry.com/artificial-intelligence-questions-answers/)
LISP Programming MCQs (http://www.sanfoundry.com/lisp-programming-questions-answers/)
Database Management MCQs (http://www.sanfoundry.com/database-management-system-questions-answers/)
Computer Network MCQs (http://www.sanfoundry.com/computer-network-questions-answers/)
Microprocessor MCQs (http://www.sanfoundry.com/microprocessors-questions-answers/)

C Programming Examples

Simple C Programs (http://www.sanfoundry.com/simple-c-programs/)
C - Arrays (http://www.sanfoundry.com/c-programming-examples-arrays/)
C - Matrix (http://www.sanfoundry.com/c-programming-examples-matrix/)
C - Strings (http://www.sanfoundry.com/c-programming-examples-strings/)
C - Bitwise Operations (http://www.sanfoundry.com/c-programming-examples-bitwise-

operations/)
C - Linked Lists (http://www.sanfoundry.com/c-programming-examples-linked-list/)
C - Stacks & Queues (http://www.sanfoundry.com/c-programming-examples-stacks/)
C - Searching & Sorting (http://www.sanfoundry.com/c-programming-examples-searching-sorting/)
C - Trees (http://www.sanfoundry.com/c-programming-examples-on-trees/)
C - Strings (http://www.sanfoundry.com/c-programming-examples-strings/)
C - File Handling (http://www.sanfoundry.com/c-programming-examples-file-handling/)
C - Mathematical Functions (http://www.sanfoundry.com/c-programming-examples-mathematical-functions/)
C - Puzzles & Games (http://www.sanfoundry.com/c-programming-examples-on-puzzles-games/)
C Programs - Recursion (http://www.sanfoundry.com/c-programming-examples-recursion/)
C Programs - No Recursion (http://www.sanfoundry.com/c-programming-examples-without-using-recursion/)

Java Algorithms

Java - Numerical Problems (http://www.sanfoundry.com/java-programming-examples-numerical-problems-algorithms/)
Java - Combinatorial Problems

(http://www.sanfoundry.com/java-programming-examples-combinatorial-problems-algorithms/)
Java - Graph Problems (http://www.sanfoundry.com/java-programming-examples-graph-problems-algorithms/)
Java - Hard Graph Problems (http://www.sanfoundry.com/java-programming-examples-hard-graph-problems-algorithms/)
Java - Computation Geometry (http://www.sanfoundry.com/java-programming-examples-computational-geometry-problems-algorithms/)
Java - Sets & Strings (http://www.sanfoundry.com/java-programming-examples-set-string-problems-algorithms/)
Java - Data-Structures (http://www.sanfoundry.com/java-programming-examples-data-structures/)
Java - Collection API Problems (http://www.sanfoundry.com/java-programming-examples-collection-api/)

C++ Algorithms

C++ - Numerical Problems (http://www.sanfoundry.com/cpp-programming-examples-numerical-problems-algorithms/)
C++ - Combinatorial Problems (http://www.sanfoundry.com/cpp-programming-examples-combinatorial-problems-algorithms/)
C++ - Graph Problems (http://www.sanfoundry.com/cpp-programming-examples-graph-problems-algorithms/)
C++ - Hard Graph Problems (http://www.sanfoundry.com/cpp-

programming-examples-hard-graph-problems-algorithms/)
C++ - Computation Geometry (http://www.sanfoundry.com/cpp-programming-examples-computational-geometry-problems-algorithms/)
C++ - Sets & Strings (http://www.sanfoundry.com/cpp-programming-examples-set-string-problems-algorithms/)
C++ - Data-Structures (http://www.sanfoundry.com/cpp-programming-examples-data-structures/)
C++ - STL Library (http://www.sanfoundry.com/cpp-programming-examples-stl/)

C Algorithms

C - Numerical Problems (http://www.sanfoundry.com/c-programming-examples-numerical-problems-algorithms/)
C - Combinatorial Problems (http://www.sanfoundry.com/c-programming-examples-combinatorial-problems-algorithms/)
C - Graph Problems (http://www.sanfoundry.com/c-programming-examples-graph-problems-algorithms/)
C - Hard Graph Problems (http://www.sanfoundry.com/c-programming-examples-hard-graph-problems-algorithms/)
C - Computation Geometry (http://www.sanfoundry.com/c-programming-examples-computational-geometry-problems-algorithms/)
C - Sets & Strings (http://www.sanfoundry.com/c-programming-examples-set-string-problems-algorithms/)

C - Data-Structures (http://www.sanfoundry.com/c-programming-examples-data-structures/)

C++ Program to Implement Binary Heap

This C++ Program demonstrates the implementation of Binary Heap.

Here is source code of the C++ Program to demonstrate Binary Heap. The C++ program is successfully compiled and run on a Linux system. The program output is also shown below.

```
1. /*
2.  * C++ Program to Implement Binary Heap
3.  */
4. #include <iostream>
5. #include <cstdlib>
6. #include <vector>
7. #include <iterator>
8. using namespace std;
9. /*
10.  * Class Declaration
11.  */
12. class BinaryHeap
13. {
14.     private:
15.         vector <int> heap;
16.         int left(int parent);
17.         int right(int parent);
18.         int parent(int child);
19.         void heapifyup(int index);
20.         void heapifydown(int index);
21.     public:
22.         BinaryHeap()
23.         {}
24.         void Insert(int element);
25.         void DeleteMin();
26.         int ExtractMin();
27.         void DisplayHeap();
```

```
28.         int Size();
29. };
30. /*
31.  * Return Heap Size
32.  */
33. int BinaryHeap::Size()
34. {
35.     return heap.size();
36. }
37. 
38. /*
39.  * Insert Element into a Heap
40.  */
41. void BinaryHeap::Insert(int element)
42. {
43.     heap.push_back(element);
44.     heapifyup(heap.size() -1);
45. }
46. /*
47.  * Delete Minimum Element
48.  */
49. void BinaryHeap::DeleteMin()
50. {
51.     if (heap.size() == 0)
52.     {
53.         cout<<"Heap is Empty"<<endl;
54.         return;
55.     }
56.     heap[0] = heap.at(heap.size() - 1);
57.     heap.pop_back();
58.     heapifydown(0);
59.     cout<<"Element Deleted"<<endl;
60. }
61. 
62. /*
63.  * Extract Minimum Element
64.  */
65. int BinaryHeap::ExtractMin()
66. {
67.     if (heap.size() == 0)
68.     {
69.         return -1;
70.     }
71.     else
72.         return heap.front();
73. }
74. 
```

```
/*
 * Display Heap
 */
void BinaryHeap::DisplayHeap()
{
    vector <int>::iterator pos = heap.begin();
    cout<<"Heap -->  ";
    while (pos != heap.end())
    {
        cout<<*pos<<" ";
        pos++;
    }
    cout<<endl;
}

/*
 * Return Left Child
 */
int BinaryHeap::left(int parent)
{
    int l = 2 * parent + 1;
    if (l < heap.size())
        return l;
    else
        return -1;
}

/*
 * Return Right Child
 */
int BinaryHeap::right(int parent)
{
    int r = 2 * parent + 2;
    if (r < heap.size())
        return r;
    else
        return -1;
}

/*
 * Return Parent
 */
int BinaryHeap::parent(int child)
{
    int p = (child - 1)/2;
    if (child == 0)
        return -1;
```

```
122.     else
123.         return p;
124. }
125. 
126. /*
127.  * Heapify- Maintain Heap Structure bottom up
128.  */
129. void BinaryHeap::heapifyup(int in)
130. {
131.     if (in >= 0 && parent(in) >= 0 && heap[parent(in)] > heap[in])
132.     {
133.         int temp = heap[in];
134.         heap[in] = heap[parent(in)];
135.         heap[parent(in)] = temp;
136.         heapifyup(parent(in));
137.     }
138. }
139. 
140. /*
141.  * Heapify- Maintain Heap Structure top down
142.  */
143. void BinaryHeap::heapifydown(int in)
144. {
145. 
146.     int child = left(in);
147.     int child1 = right(in);
148.     if (child >= 0 && child1 >= 0 && heap[child] > heap[child1])
149.     {
150.        child = child1;
151.     }
152.     if (child > 0)
153.     {
154.         int temp = heap[in];
155.         heap[in] = heap[child];
156.         heap[child] = temp;
157.         heapifydown(child);
158.     }
159. }
160. 
161. /*
162.  * Main Contains Menu
163.  */
164. int main()
165. {
166.     BinaryHeap h;
167.     while (1)
168.     {
```

```
        cout<<"------------------"<<endl;
        cout<<"Operations on Heap"<<endl;
        cout<<"------------------"<<endl;
        cout<<"1.Insert Element"<<endl;
        cout<<"2.Delete Minimum Element"<<endl;
        cout<<"3.Extract Minimum Element"<<endl;
        cout<<"4.Print Heap"<<endl;
        cout<<"5.Exit"<<endl;
        int choice, element;
        cout<<"Enter your choice: ";
        cin>>choice;
        switch(choice)
        {
        case 1:
            cout<<"Enter the element to be inserted: ";
            cin>>element;
            h.Insert(element);
            break;
        case 2:
            h.DeleteMin();
            break;
        case 3:
            cout<<"Minimum Element: ";
            if (h.ExtractMin() == -1)
            {
                cout<<"Heap is Empty"<<endl;
            }
            else
                cout<<"Minimum Element:  "<<h.ExtractMin()<<endl;
            break;
        case 4:
            cout<<"Displaying elements of Hwap:  ";
            h.DisplayHeap();
            break;
        case 5:
            exit(1);
        default:
            cout<<"Enter Correct Choice"<<endl;
        }
    }
    return 0;
}
```

```
$ g++ heap.cpp
$ a.out
```

```
/*
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 1
Enter the element to be inserted: 7
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 4
Displaying elements of Hwap:  Heap -->  7
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 1
Enter the element to be inserted: 4
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 4
Displaying elements of Hwap:  Heap -->  4 7
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
```

```
4.Print Heap
5.Exit
Enter your choice: 1
Enter the element to be inserted: 9
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 4
Displaying elements of Hwap:  Heap -->  4 7 9
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 1
Enter the element to be inserted: 3
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 4
Displaying elements of Hwap:  Heap -->  3 4 9 7
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 3
Minimum Element: Minimum Element:  3
------------------
Operations on Heap
------------------
```

```
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 1
Enter the element to be inserted: 5
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 4
Displaying elements of Hwap:  Heap -->  3 4 9 7 5
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 1
Enter the element to be inserted: 11
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 4
Displaying elements of Hwap:  Heap -->  3 4 9 7 5 11
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 1
Enter the element to be inserted: 2
```

```
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 4
Displaying elements of Hwap:  Heap -->  2 4 3 7 5 11 9
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 2
Element Deleted
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 4
Displaying elements of Hwap:  Heap -->  3 4 11 7 5 9
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 3
Minimum Element: Minimum Element:  3
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
```

```
5.Exit
Enter your choice: 2
Element Deleted
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 4
Displaying elements of Hwap:  Heap -->  4 5 11 7 9
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 3
Minimum Element: Minimum Element:  4
------------------
Operations on Heap
------------------
1.Insert Element
2.Delete Minimum Element
3.Extract Minimum Element
4.Print Heap
5.Exit
Enter your choice: 5

------------------
(program exited with code: 0)
Press return to continue
```

Sanfoundry Global Education & Learning Series – 1000 C++ Programs.

If you wish to look at all C++ Programming examples, go to C++ Programs (http://www.sanfoundry.com/).

If you liked this C++ Program, kindly share, recommend or like below!

Read Next:

1. C++ Program to Implement Heap (http://www.sanfoundry.com/cpp-program-implement-heap/)
2. Java Program to Implement Binary Heap (http://www.sanfoundry.com/java-program-implement-binary-heap/)
3. Java Program to Implement D-ary-Heap (http://www.sanfoundry.com/java-program-implement-d-ary-heap/)
4. Java Program to Implement Ternary Heap (http://www.sanfoundry.com/java-program-implement-ternary-heap/)
5. C++ Program to Implement Fibonacci Heap (http://www.sanfoundry.com/cpp-program-implement-fibonacci-heap/)

c=33&key=10e5d4b2f54ff3d3fc27c4ce6d6d086c&utm_source=Taboola&utm_medium=CPC&utm_content=text-32-HH-05-01-r-1r2&utm_campaign=HH-Taboola-US)

The #1 Reason Amateur Golfers Can't Drive 200+ Yards...

Hank Haney University

(http://scout.golferstrust.com/scout/base.php?c=33&key=10e5d4b2f54ff3d3fc27c4ce6d6d086c&utm_source=Taboola&utm_medium=CPC&utm_content=text-32-HH-05-01-r-1r2&utm_campaign=HH-Taboola-US)
(http://www.onesmartpenny.com/landers/pages/shark-tanks-barbara-corcorans-advice-on-mortgage.html?utm_source=taboola&utm_medium=%7Bsite%7D&utm_term=revealsbrill_corcoran2_st&utm_adgroup=desktop_st&utm_medium=sanfoundry)

"Shark Tank" Star Reveals Brilliant Mortgage Payoff Tip

Bills.com

(http://www.onesmartpenny.com/landers/pages/shark-tanks-barbara-corcorans-advice-on-mortgage.html?utm_source=taboola&utm_medium=%7Bsite%7D&utm_term=revealsbrill_corcoran2_st&utm_adgroup=desktop_st&utm_medium=sanfoundry)
(http://www.nextadvisor.com/blog/2013/11/06/top-7-credit-card-offers-for-those-with-excellent-credit/?kw=tbla_dsk_mpa10&&site=sanfoundry)

7 Outrageous Credit Cards For Those Of Us That Have Excellent Credit

NextAdvisor

(http://www.nextadvisor.com/blog/2013/11/06/top-7-credit-card-offers-for-those-with-excellent-credit/?kw=tbla_dsk_mpa10&&site=sanfoundry)
(http://dishwashers.reviewed.com/news/scientists-conclude-we-dont-load-dishwashers-correctly?utm_source=TB_paid&utm_medium=cpc&utm_campaign=sanfoundry)

Scientists Conclude We Don't Load Dishwashers Correctly

Reviewed.com

(http://dishwashers.reviewed.com/news/scientists-conclude-we-dont-load-dishwashers-correctly?utm_source=TB_paid&utm_medium=cpc&utm_campaign=sanfoundry)
(http://drive.odometer.com/click.php?net=tab&lst=www.odometer.com/lifestyle/9545/27-bizarre-aircraft-we-cant-believe-existhow-do-some-of-these-even-fly&drv=9545-tde&utm_term=sanfoundry)

27 Bizarre Aircraft We Can't Believe Exist (How Do Some Of These Even Fly?)

Odometer

(http://drive.odometer.com/click.php?net=tab&lst=www.odometer.com/lifestyle/9545/27-bizarre-aircraft-we-cant-believe-existhow-do-some-of-these-even-fly&drv=9545-tde&utm_term=sanfoundry)
(http://sp1.convertro.com/trax/short/plated/1/?redirect_to=http%3A%2F%2Fny.eater.com%2F2014%2F9%2F29%2F6844057%2Fthe-new-diy-dinner-hack&cvosrc=content+marketing.taboola.eater1&utm_source=taboola&utm_site=sanfoundry&utm_title=Millennials+Are+Ditching+Delivery+for+This+Dinner+Hack&utm_thumbnail=http%3A%2F%2Fcdn.taboolasyndication.com%2Flibtrc%2Fstatic%2

Fthumbnails%2F9044c567dd6b62e2dce4b882b2b5c8c5.png&cvo_site=sanfoundry&cvo_thumbnail=http%3A%2F%2Fcdn.taboolasyndication.com%2Flibtrc%2Fstatic%2Fthumbnails%2F9044c567dd6b62e2dce4b882b2b5c8c5.png&cvo_title=Millennials+Are+Ditching+Delivery+for+This+Dinner+Hack)

Millennials Are Ditching Delivery for This Dinner Hack

Eater for Plated

(http://sp1.convertro.com/trax/short/plated/1/?redirect_to=http%3A%2F%2Fny.eater.com%2F2014%2F9%2F29%2F6844057%2Fthe-new-diy-dinner-hack&cvosrc=content+marketing.taboola.eater1&utm_source=taboola&utm_site=sanfoundry&utm_title=Millennials+Are+Ditching+Delivery+for+This+Dinner+Hack&utm_thumbnail=http%3A%2F%2Fcdn.taboolasyndication.com%2Flibtrc%2Fstatic%2Fthumbnails%2F9044c567dd6b62e2dce4b882b2b5c8c5.png&cvo_site=sanfoundry&cvo_thumbnail=http%3A%2F%2Fcdn.taboolasyndication.com%2Flibtrc%2Fstatic%2Fthumbnails%2F9044c567dd6b62e2dce4b882b2b5c8c5.png&cvo_title=Millennials+Are+Ditching+Delivery+for+This+Dinner+Hack)
(http://www.answers.com/browse/click.php?source=tb&param3=www.answers.com/entertainment/movies/10-awesome-movies-that-slipped-through-the-cracks&param4=tb-us-de-enter&param1=sanfoundry&param2=10+Awesome+Movies+That+Slipped+Through+the+Cracks&param5=http%3A%2F%2Ffile.answcdn.com%2Fanswcld%2Fimage%2Fupload%2Fw_560%252Ch_293%252Cg_faces%252Cc_fill%2Fv1%2Ftk%2Fview%2Fcew%2F3%2F6%2F6%2F9%2F6%2F3669691892%2F0ab3b17ad9a902fe27c8cae5c472b75283a8238b.jpeg)

10 Awesome Movies That Slipped Through the Cracks

Answers

(http://www.answers.com/browse/click.php?source=tb&param3=www.answers.com/entertainment/movies/10-awesome-movies-that-slipped-through-the-cracks&param4=tb-us-de-enter&param1=sanfoundry&param2=10+Awesome+Movies+That+Slipped+Through+the+Cracks&param5=http%3A%2F%2Ffile.answcdn.com%2Fanswcld%2Fimage%2Fupload%2Fw_560%252Ch_293%252Cg_faces%252Cc_fill%2Fv1%2Ftk%2Fview%2Fcew%2F3%2F6%2F6%2F9%2F6%2F3669691892%2F0ab3b17ad9a902fe27c8cae5c472b75283a8238b.jpeg)

Manish Bhojasia (http://www.sanfoundry.com/about/), a technology veteran with 18+ years @ Cisco & Wipro, is Founder and CTO at Sanfoundry. He is Linux Kernel Developer and SAN Architect and is passionate about competency developments in these areas. He lives in Bangalore and delivers focused training sessions to IT professionals in Linux Kernel, Linux Debugging, Linux Device Drivers, Linux Networking, Linux Storage & Cluster Administration, Advanced C Programming, SAN Storage Technologies, SCSI Internals and Storage Protocols such as iSCSI & Fiber Channel. Stay connected with him below:
Google+ (http://google.com/+ManishBhojasia) | Facebook (http://www.facebook.com/sanfoundry) | Twitter (http://www.twitter.com/sanfoundry) | LinkedIn (http://www.linkedin.com/in/manishbhojasia)

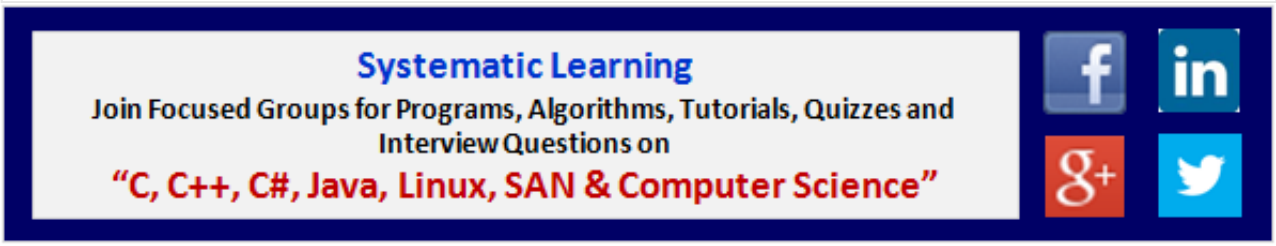
Name

*

Email Address

*

Subscribe



(http://www.sanfoundry.com/technology-focussed-groups/)

Best Training Courses

SAN Training I (http://www.sanfoundry.com/san-storage-area-networks-training/)
SAN Administration II (http://www.sanfoundry.com/san-administration-training-course/)
Linux Administration (http://www.sanfoundry.com/linux-administration-training/)
Advanced C Programming (http://www.sanfoundry.com/advanced-c-programming-training/)
Linux Debugging (http://www.sanfoundry.com/training-on-linux-debugging-techniques/)
Linux Programming (http://www.sanfoundry.com/training-on-linux-internals-systems/)
Linux IPCs Programming (http://www.sanfoundry.com/training-interprocess-

communication/)
Network Programming (http://www.sanfoundry.com/training-socket-network-programming/)
Linux Multithreading (http://www.sanfoundry.com/training-multithreaded-parallel/)
Linux Kernel Programming (http://www.sanfoundry.com/linux-kernel-internals-training/)
Linux Kernel Debugging (http://www.sanfoundry.com/linux-kernel-debugging-training/)
Device Drivers Basics (http://www.sanfoundry.com/training-on-linux-device-drivers/)
Linux Block Drivers (http://www.sanfoundry.com/linux-block-device-drivers-training/)
Linux Network Drivers (http://www.sanfoundry.com/linux-network-device-drivers-training/)
Linux PCI Drivers (http://www.sanfoundry.com/linux-pci-device-drivers-training/)
Linux USB Drivers (http://www.sanfoundry.com/linux-usb-device-drivers-training/)
Linux Video Drivers (http://www.sanfoundry.com/linux-video-device-drivers-training/)
Linux Audio Drivers (http://www.sanfoundry.com/linux-audio-device-drivers-training/)
I2C Driver Training (http://www.sanfoundry.co

m/linux-i2c-device-drivers-training/)

Technical Career

Developer Tracks (http://www.sanfoundry.com/salary-50l/)
SAN Developer (http://www.sanfoundry.com/san-storage-developer-training-courses/)
Linux System Developer (http://www.sanfoundry.com/linux-developer-training-courses-jobs/)

Linux Kernel/Driver Developer (http://www.sanfoundry.com/linux-device-driver-developer-training/)
Linux Network Developer (http://www.sanfoundry.com/linux-network-developer-training/)
Mentoring (http://www.sanfoundry.com/professional-mentoring-coaching-career-guidance-cto/)
Productivity (http://www.sanfoundry.com/programming-discipline-and-software/)
Coding-Style (http://www.sanfoundry.com/c-coding-guidelines-developers/)
GDB Assignment (http://www.sanfoundry.com/gdb-example-tutorial/)
Feedback (http://www.sanfoundry.com/feedback/)

From The Web

ent=text-32-HH-05-01-r-1r2&utm_campaign=HH-Taboola-US)

The #1 Reason Amateur Golfers Can't Drive 200+ Yards...

Hank Haney University

(http://scout.golferstrust.com/scout/base.php?c=33&key=10e5d4b2f54ff3d3fc27c4ce6d6d086c&utm_source=Taboola&utm_medium=CPC&utm_content=text-32-HH-05-01-r-1r2&utm_campaign=HH-Taboola-US)

(http://www.onesmartpenny.com/landers/pages/shark-tanks-barbara-corcorans-advice-on-mortgage.html?utm_source=taboola&utm_medium=%7Bsite%7D&utm_term=revealsbrill_corcoran2_st&utm_adgroup=desktop_st&utm_medium=sanfoundry)

"Shark Tank" Star Reveals Brilliant Mortgage Payoff Tip

Bills.com

(http://www.onesmartpenny.com/landers/pages/shark-tanks-barbara-corcorans-advice-on-mortgage.html?utm_source=taboola&utm_medium=%7Bsite%7D&utm_term=revealsbrill_corcoran2_st&utm_adgroup=desktop_st&utm_medium=sanfoundry)

(http://www.nextadvisor.com/blog/2013/11/06/top-7-credit-card-offers-for-those-with-excellent-credit/?kw=tbla_dsk_mpa10&&site=sanfoundry)

7 Outrageous Credit Cards For Those Of Us That Have Excellent…

NextAdvisor

(http://www.nextadvisor.com/blog/2013/11/06/top-7-credit-card-offers-for-those-with-excellent-credit/?kw=tbla_dsk_mpa10&&site=sanfoundry)

ent=thumbnails-a:Right Rail Thumbnails:)

ent=thumbnails-a:Right Rail Thumbnails:)