

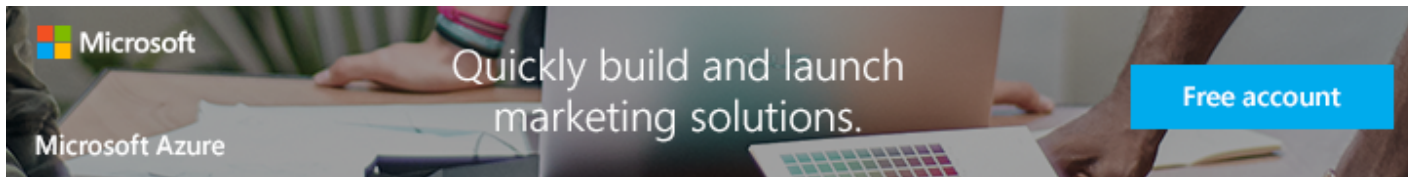
[x Dismiss](#)

## Join the Stack Overflow Community

Stack Overflow is a community of 7.0 million programmers, just like you, helping each other.  
Join them; it only takes a minute:

[Sign up](#)

## Two Dimensional Vector push\_back



I have a two dimensional vector A made up of other vectors B

```
vector < vector < int >> A
```

```
vector < int > B
```

I use the `push_back` function to populate B.

```
B.push_back(1);  
B.push_back(2);  
B.push_back(3);
```

After that vector is populated, I use `push_back` again to populate A with B

```
A.push_back(B)
```

This is done several times so that A eventually results in a vector containing several other vectors looking like:

```
A { {1, 2, 3}, {2, 2, 2}, {8, 9, 10} }
```

How can I make a call to a specific index in A and then continue to add to the vector so that the output would be similar to

```
A { {1, 2, 3}, {2, 2, 2, 4, 5, 6}, {8, 9, 10} }
```

Something along the lines of

```
A[2].push_back(4);  
A[2].push_back(5);  
A[2].push_back(6);
```

c++ vector

asked May 1 '13 at 4:21



Barry Tormey

1,188 2 13 40

## 2 Answers

What you have is correct except that indexes start at 0, so it should be `A[1].push_back(4);` and not 2.

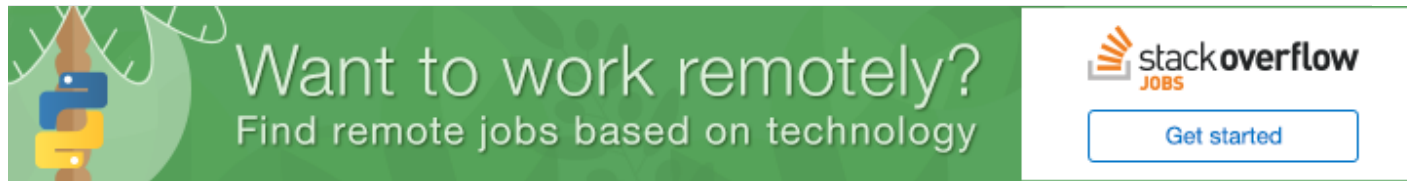
answered May 1 '13 at 4:24



Jesse Good

33.2k 7 65 121

- 1 @BarryTormey: Could you post the full example? [Here is an example of your code](#) that works fine. – Jesse Good May 1 '13 at 4:40



```
A[2].push_back(4);  
A[2].push_back(5);  
A[2].push_back(6);
```

Should work perfectly fine. Except if you want the second element then you'll need to use `a[1]` as vectors are 0 based.

answered May 1 '13 at 4:26



[iambeanie](#)

195 7