```
        normalChar[i] = ch ;

    for (i=0 ; i<26 ; i++)
        cipherChar[i] = normalChar[(i+3)%26];
```

الآن ، لكي نقوم بالتشفير ، نفرض أننا أدخلنا النص المراد تشفيره في المصفوفة inputChar ، ونريد النص المشفر يكون في المصفوفة outputChar :

```
for (i=0 ; i<length ; i++)
{
    if ( inputChar[i] == ' ' )            // if it's space , ignore it
    {
            outputChar[i] = ' ' ;
            continue;
    }

    int x = (int)inputChar[i] ;           // take the character as number

    x = x -  65 ;                         // make the number from 0-25

outputChar[i] = cipherChar[x] ;   // assign the cipher char to output
    cout << outputChar[i]  ;
}
```

داخل الحلقة ، قمنا بتحويل الحرف الأول في المصفوفة إلى نوع int ، وهكذا تحصلنا على قيمه الأسكي الخاص بالحرف الأول ، ولنفرض أنه الحرف B فقيمه الأسكي هنا في هذه الحالة 66 . بعدها نطرح من هذه القيمة 65 ، لكي نرجع قيمه المتغير إلى رقم مداه من 0 إلى 25 ، وهنا يكون هذا المتغير هو الindex الخاص بمصفوفة الحروف المشفرة cipherChar التي قمنا بتهيئتها في البداية .

لفك التشفير ، لنفرض أن النص المشفر موجود داخل outputChar ونريد النص الأصلي في newNormal

```
for (int i=0 ; i<length ; i++)
{
    if ( outputChar[i] == ' ' )
    {
         newNormal[i] = ' ' ;
         continue;
    }

    int x = (int)outputChar[i] ;  // get cipher char , convert to int

    x = x - 65 ;                    // return it to number
    x = x - key ;                  // sub from it the key

    // if it's negative make it from the begin of array
    if ( x<0)   x = 26-abs(x) ;

    newNormal[i] = normalChar[x] ;
    cout << newNormal[i] ;
}
```


PDF created with pdfFactory Pro trial version www.pdffactory.com