

[Courseware \(/courses/UTAustinX/UT.6.01x/1T2014/courseware\)](/courses/UTAustinX/UT.6.01x/1T2014/courseware)

[Course Info \(/courses/UTAustinX/UT.6.01x/1T2014/info\)](/courses/UTAustinX/UT.6.01x/1T2014/info)

[Discussion \(/courses/UTAustinX/UT.6.01x/1T2014/discussion/forum\)](/courses/UTAustinX/UT.6.01x/1T2014/discussion/forum)

[Progress \(/courses/UTAustinX/UT.6.01x/1T2014/progress\)](/courses/UTAustinX/UT.6.01x/1T2014/progress)

[Questions \(/courses/UTAustinX/UT.6.01x/1T2014/a3da417940af4ec49a9c02b3eae3460b/\)](/courses/UTAustinX/UT.6.01x/1T2014/a3da417940af4ec49a9c02b3eae3460b/)

[Syllabus \(/courses/UTAustinX/UT.6.01x/1T2014/a827a8b3cc204927b6efaa49580170d1/\)](/courses/UTAustinX/UT.6.01x/1T2014/a827a8b3cc204927b6efaa49580170d1/)

Example 11.1. You are given a subroutine, `UART_OutChar`, which outputs one ASCII character. Design a function that outputs a 32-bit unsigned integer.

Solution: We will solve this iteratively. As always, we ask “what is our starting point?”, “how do we make progress?”, and “when are we done?” The input, n , is a 32-bit unsigned number, and we are done when 1 to 10 ASCII characters are displayed, representing the value of n . Figure 11.9 demonstrates the successive refinement approach to solving this problem iteratively. The iterative solution has three phases: initialization, creation of digits, and output of the ASCII characters. The digits are created from the remainders occurring by dividing the input, n by 10. To get all the digits we divide by 10 until the quotient is 0. Because the digits are created in the opposite order, each digit will be saved in a buffer during the creation phase and retrieved from the buffer during the output stage. The counter is needed so the output stage knows how many digits are in the buffer.

Help

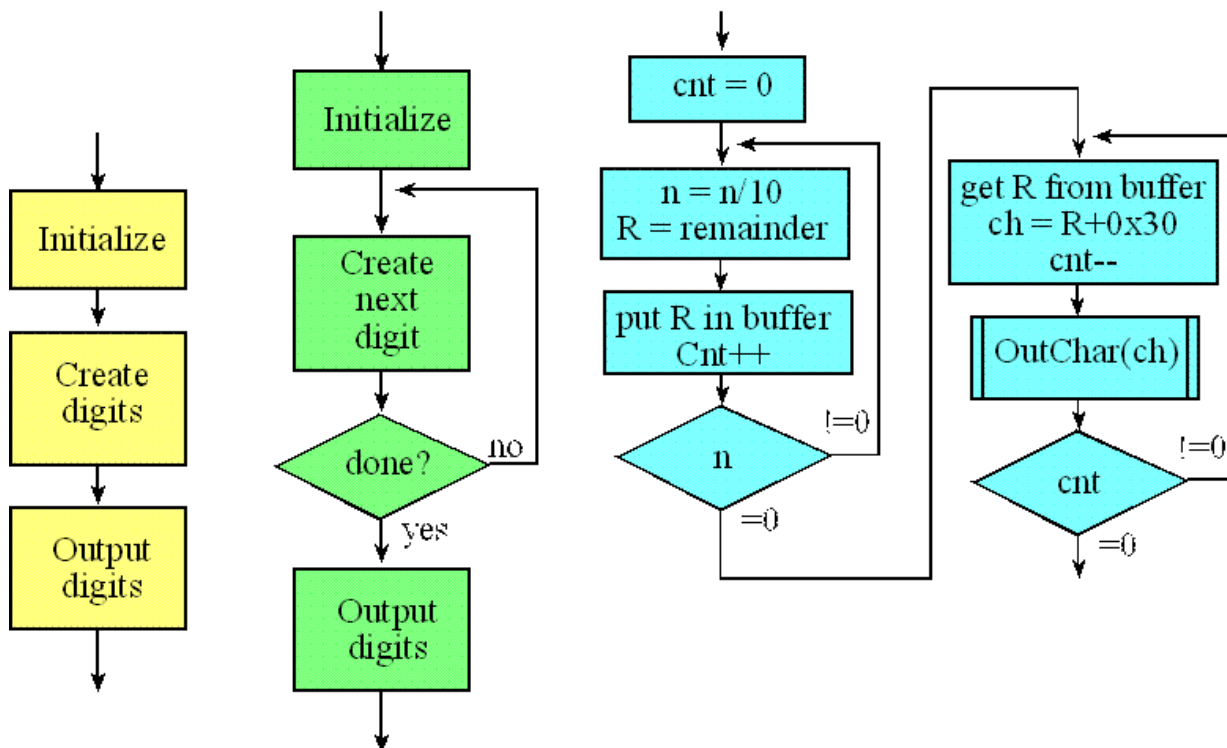


Figure 11.9. Successive refinement method for the iterative solution.

The iteration solution requires two loops; the first loop determines the digits in opposite order, and the second loop outputs the digits in proper order.

```
// iterative method
void OutUDec(unsigned long n){
    unsigned cnt=0;
    unsigned char buffer[11];
    do{
        buffer[cnt] = n%10; // digit
        n = n/10;
        cnt++;
    }
    while(n); // repeat until n==0
    for(; cnt; cnt--){
        OutChar(buffer[cnt-1]+'0');
    }
}
```

Program 11.7. Iterative implementation of output decimal.

Help



About (<https://www.edx.org/about-us>) Jobs (<https://www.edx.org/jobs>)
Press (<https://www.edx.org/press>) FAQ (<https://www.edx.org/student-faq>)
Contact (<https://www.edx.org/contact>)



EdX is a non-profit created by founding partners Harvard and MIT whose mission is to bring the best of higher education to students of all ages anywhere in the world, wherever there is Internet access. EdX's free online MOOCs are interactive and subjects include computer science, public health, and artificial intelligence.



(<http://www.meetup.com/edX-Global-Community/>)



(<http://www.facebook.com/EdxOnline>)



(<https://twitter.com/edXOnline>)



(<https://plus.google.com/108235383044095082735/posts>)



(<http://youtube.com/user/edxonline>)

© 2014 edX, some rights reserved.

Terms of Service and Honor Code -
Privacy Policy (<https://www.edx.org/edx-privacy-policy>)