Ugly Numbers

```
#include <stdio.h>
#include <vector>

std::vector<long long int> ans;
void uglyNumber(int n) {
    ans.push_back(1);
    int a = 0, b = 0, c = 0;
    for(int i = 1; i < n; i++){
        ans.push_back(std::min(ans[a] * 2, std::min(ans[b] * 3, ans[c] * 5)));
        if(ans[i] == ans[a] * 2)
            a++;
        if(ans[i] == ans[b] * 3)
            b++;
        if(ans[i] == ans[c] * 5)
            c++;
    }
}

int main() {
                // 提前计算好所有的值,直接返回,避免重复计算
    uglyNumber(10001);
    int t = 0, n = 0;
    scanf("%d", &t);
    while (t--) {
        scanf("%d", &n);
        printf("%lld\n", ans[n - 1]);
    }
    return 0;
}
```

思路:

```
1 Declare an array for ugly numbers:  ugly[n]
2 Initialize first ugly no:  ugly[0] = 1
3 Initialize three array index variables i2, i3, i5 to point to
   1st element of the ugly array:
        i2 = i3 = i5 =0;
4 Initialize 3 choices for the next ugly no:
         next_mulitple_of_2 = ugly[i2]*2;
         next_mulitple_of_3 = ugly[i3]*3
         next_mulitple_of_5 = ugly[i5]*5;
5 Now go in a loop to fill all ugly numbers till 150:
For (i = 1; i < 150; i++ )
{
    /* These small steps are not optimized for good
```

```
       readability. Will optimize them in C program */
    next_ugly_no  = Min(next_mulitple_of_2,
                        next_mulitple_of_3,
                        next_mulitple_of_5);

    ugly[i] =  next_ugly_no

    if (next_ugly_no  == next_mulitple_of_2)
    {
        i2 = i2 + 1;
        next_mulitple_of_2 = ugly[i2]*2;
    }
    if (next_ugly_no  == next_mulitple_of_3)
    {
        i3 = i3 + 1;
        next_mulitple_of_3 = ugly[i3]*3;
     }
     if (next_ugly_no  == next_mulitple_of_5)
     {
        i5 = i5 + 1;
        next_mulitple_of_5 = ugly[i5]*5;
     }

}/* end of for loop */
6.return next_ugly_no
```