Euler problem 04

于船长

书山有路勤为径，学海无涯苦作舟

本期内容

一. 题目讲解

二. 代码演示

三. 程序优化

一. 题目讲解

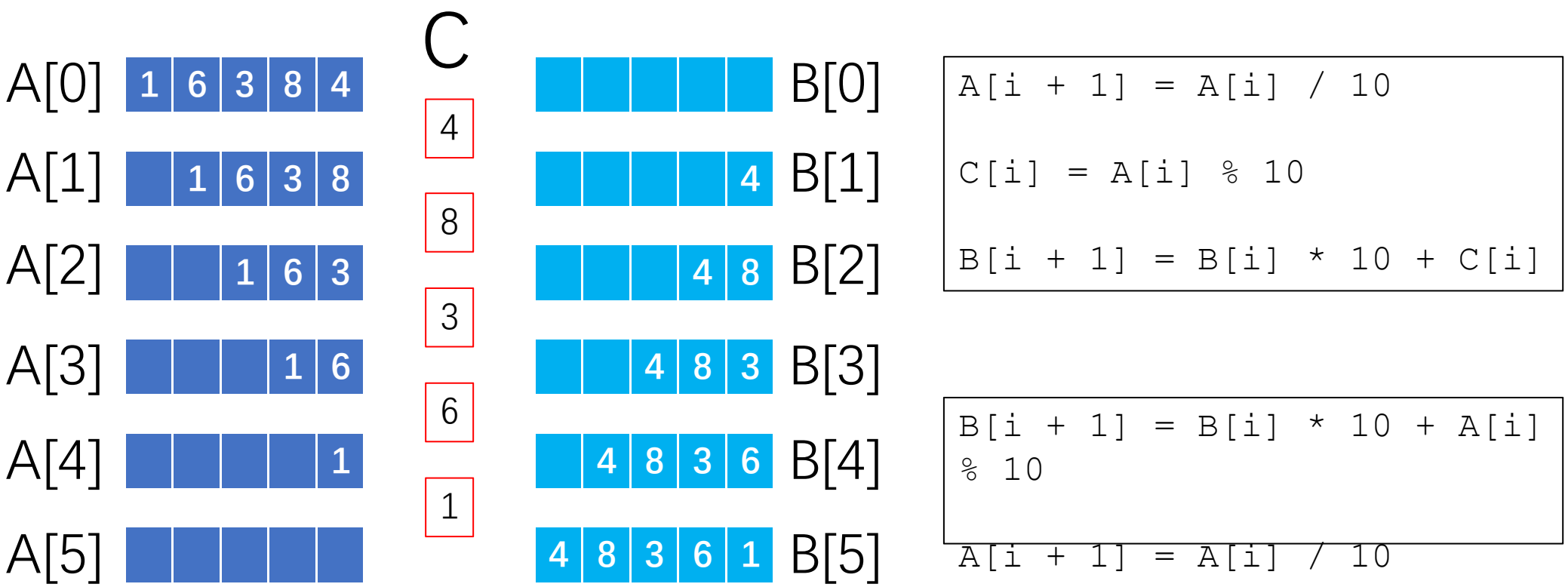
一. 题目讲解

Largest palindrome product
A palindromic number reads the same both ways. The largest palindrome made from the product of two 2-digit numbers is 9009 = 91 × 99.
Find the largest palindrome made from the product of two 3-digit numbers.

最大回文乘积
回文数是指从前往后读和从后往前读都一样的数。由两个2位数相乘得到的回文数中，最大的是9009 = 91 × 99。
求最大的由两个3位数相乘得到的回文数。

一. 题目讲解

整数翻转




```
A[i + 1] = A[i] / 10

C[i] = A[i] % 10

B[i + 1] = B[i] * 10 + C[i]
```

```
B[i + 1] = B[i] * 10 + A[i]
% 10

A[i + 1] = A[i] / 10
```

二. 代码演示

二. 代码演示

```c
#include <stdio.h>

int is_val(int n) {
    int temp = 0, x = n;
    while (x) {
        temp = temp * 10 + x % 10;
        x /= 10;
    }
    return temp == n;
}

int main() {
    int ans = 0;
    for (int a = 100; a < 1000; a++) {
        for (int b = 100; b < 1000; b++) {
            if (is_val(a * b) && a * b > ans) {
                ans = a * b;
            }
        }
    }
    printf("%d\n", ans);
    return 0;
}
```

Q:程序能否再优化一下?

三. 程序优化

三. 程序优化

```c
#include <stdio.h>

int is_val(int x, int base) {
    int tmp = x, sum = 0;
    while (x) {
        sum = sum * base + x % base;
        x /= base;
    }
    return sum == tmp;
}

int main() {
    int ans = 0;
    for (int a = 100; a < 1000; a++) {
        for (int b = ans / a + 1; b <= a; b++) {
            if (is_val(a * b, 10) && ans < a * b) {
                ans = a * b;
                printf("%d * %d = %d\n", a, b, ans);
            }
        }
    }
    printf("%d\n", ans);
    return 0;
}
```