# Ch 4.2 Integer Representations and Algorithms (R)

*Basically everything in Topic 1 Number bases with a little extra/advanced stuff.

*Skip everything I already knew

> ⭐ Let $b$ be an integer greater than 1. Then if $n$ is a positive integer, it can be expressed uniquely in the form
> $n = a_k b^k + a_{k-1} b^{k-1} + \cdots + a_1 b + a_0$,
> where $k$ is a nonnegative integer, $a_0, a_1, \ldots, a_k$ are nonnegative integers less than $b$, and $a_k \neq 0$.

**Conversion Between Binary, Octal, and Hexadecimal Expansions**

▼ Example 7

Find the octal and hexadecimal expansions of $(11\ 1110\ 1011\ 1100)_2$ and the binary expansions of $(765)_8$ and $(A8D)_{16}$.

*Solution*:

To convert binary into octal notation, we group the binary digits into blocks of three, adding initial zeros at the start of the leftmost block if necessary.

$(11\ 1110\ 1011\ 1100)_2 \ \Rightarrow\ 011\ 111\ 010\ 111\ 100 \ \Rightarrow 3\ 7\ 2\ 7\ 4$

Therefore, $(11\ 1110\ 1011\ 1100)_2 = (37274)_8$

To convert binary into hexadecimal notation, we group the binary digits into blocks of four, adding initial zeros at the start of the leftmost block if necessary.

$(11\ 1110\ 1011\ 1100)_2 \ \Rightarrow\ 0011\ 1110\ 1011\ 1100 \ \Rightarrow\ 3\ E\ B\ C$

Therefore, $(11\ 1110\ 1011\ 1100)_2 = (3EBC)_{16}$

To convert octal into binary notation, we replace each octal digit by a block of three binary digits.

$(765)_8 \ \Rightarrow\ 111\ 110\ 101$

Therefore, $(765)_8 = (1\ 1111\ 0101)_2$

To convert hexadecimal into binary notation, we replace each hexadecimal digit by a block of four binary digits.

$(A8D)_{16} \ \Rightarrow\ 1010\ 1000\ 1101$

Therefore, $(A8D)_{16} = (1010\ 1000\ 1101)_2$.

**ALGORITHM 5 Fast Modular Exponentiation.**

**procedure** *modular exponentiation*($b$: integer, $n = (a_{k-1}a_{k-2} \ldots a_1 a_0)_2$,
  $m$: positive integers)
$x := 1$
*power* $:= b$ **mod** $m$
**for** $i := 0$ **to** $k - 1$
  **if** $a_i = 1$ **then** $x := (x \cdot power)$ **mod** $m$
  *power* $:= (power \cdot power)$ **mod** $m$
**return** $x\{x$ equals $b^n$ **mod** $m\}$

▼ Example 12  Use Algorithm 5 to find $3^{644}$ **mod** 645.

*Solution*:

Algorithm 5 initially sets $x = 1$ and *power* = 3 **mod** 645 $= 3$. In the computation of $3^{644}$ **mod** 645, this algorithm determines $3^{2^j}$ **mod** 645 for $j = 1, 2, \ldots, 9$ by successively squaring and reducing modulo 645. If $a_j = 1$ (where $a_j$ is the bit in the $j$th position in the binary expansion of 644), which is $(1010000100)_2$, it multiplies the current value of $x$ by $3^{2^j}$ **mod** 645 and reduces the result modulo 645. Here are the steps used:

$i = 0$: Because $a_0 = 0$, we have $x = 1$ and $power = 3^2$ **mod** $645 = 9$ **mod** $645 = 9$;

$i = 1$: Because $a_1 = 0$, we have $x = 1$ and $power = 9^2$ **mod** $645 = 81$ **mod** $645 = 81$;

$i = 2$: Because $a_2 = 1$, we have $x = 1 \cdot 81$ **mod** $645 = 81$ and $power = 81^2$ **mod** $645 = 6561$ **mod** $645 = 111$;

$i = 3$: Because $a_3 = 0$, we have $x = 81$ and $power = 111^2$ **mod** $645 = 12{,}321$ **mod** $645 = 66$;

$i = 4$: Because $a_4 = 0$, we have $x = 81$ and $power = 66^2$ **mod** $645 = 4356$ **mod** $645 = 486$;

$i = 5$: Because $a_5 = 0$, we have $x = 81$ and $power = 486^2$ **mod** $645 = 236{,}196$ **mod** $645 = 126$;

$i = 6$: Because $a_6 = 0$, we have $x = 81$ and $power = 126^2$ **mod** $645 = 15{,}876$ **mod** $645 = 396$;

$i = 7$: Because $a_7 = 1$, we find that $x = (81 \cdot 396)$ **mod** $645 = 471$ and $power = 396^2$ **mod** $645 = 156{,}816$ **mod** $645 = 81$;

$i = 8$: Because $a_8 = 0$, we have $x = 471$ and $power = 81^2$ **mod** $645 = 6561$ **mod** $645 = 111$;

$i = 9$: Because $a_9 = 1$, we find that $x = (471 \cdot 111)$ **mod** $645 = 36$.