

اصول و مبانی برنامه‌نویسی



مجید شبیری

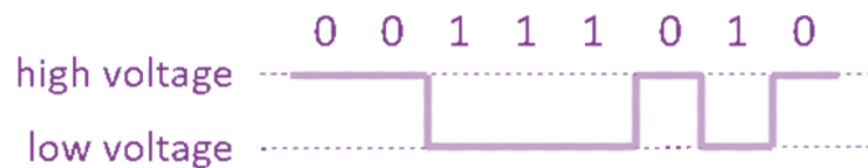
کارشناسی ارشد IT، گرایش شبکه
از دانشگاه صنعتی امیرکبیر



سیستم‌های اعداد و کدینگ اطلاعات

- (۱) سیستم اعداد چیست؟
- (۲) سیستم اعداد دسیمال (Decimal)
- (۳) سیستم اعداد باینری (Binary)
- (۴) سیستم اعداد اکتال (Octal)
- (۵) سیستم اعداد هگز (Hexadecimal)
- (۶) تبدیلات در سیستم اعداد
- (۷) سیستم کد گذاری چیست؟
- (۸) کد اسکی (ASCII)
- (۹) کد ایسکی (ISCI)
- (۱۰) یونیکد (Unicode)
- (۱۱) واحدهای حافظه در کامپیوتر

- کامپیوتر ماشینی است که تنها قادر به تشخیص سیگنال‌های الکتریکی می‌باشد.
- برنامه‌نویس چطور می‌تواند **دستورات** (دستورالعمل) خود را به شکل سیگنال‌های الکتریکی به کامپیوتر انتقال دهد؟!
 - ساده‌ترین سیستم برای تشخیص دستورالعمل‌ها به کمک سیگنال‌های الکتریکی، **سیستم دو حالت on-off** است.
 - **حالت on**: با سطح ولتاژ 1 نمایش داده می‌شود.
 - **حالت off**: با سطح ولتاژ 0 نمایش داده می‌شود (البته نه صفر واقعی بلکه سطح ولتاژ کمتر از 1)
- این سیستم منجر به ابداع یک **سیستم عددی** به نام **سیستم اعداد باینری** (دودویی) شد.



- به تکنیک‌های نمایش اعداد و کار با اعداد، سیستم اعداد گفته می‌شود.
- در سیستم اعداد دودویی، همه داده‌های ورودی و دستورالعمل‌ها به شکل 0,1 به کامپیوتر انتقال داده می‌شود.
- سیستم عددی باینری برای کاربر یک سیستم سطح بالا (بالا تر از سیگنال) محسوب می‌شود.
- ولی همچنان کار کردن با 0,1 برای برنامه‌نویس ساده نبوده و محاسبات و تبدیلات متعدد باینری برای کاربر، زمانبر است.
- بنابراین، برای راحتی بیشتر برنامه‌نویسان، سیستم‌های عددی دیگری نیز ابداع شد.

○ سیستم اعداد **باینری** (مبنای ۲)

○ سیستم اعداد **اکتال** (مبنای ۸)

○ سیستم اعداد **دسیمال** (مبنای ۱۰)

○ سیستم اعداد **هگزا دسیمال** (مبنای ۱۶)

■ پیاده‌سازی سیستم‌های عددی غیر باینری، نیاز به **سطوح ولتاژ بیشتر** داشته و برای حفظ سادگی، فقط سیستم باینری در کامپیوتر پیاده‌سازی شده است.

■ سیستم باینری، امروزه به عنوان سیستم اعداد اصلی کامپیوترها مورد استفاده قرار گرفته و داده‌های نمایش داده شده در سیستم‌های عددی غیر باینری نیز در زمان کامپایل به سیستم باینری (0,1) ترجمه می‌شوند.

- سیستم اعداد دسیمال، بر مبنای ۱۰ بوده و شامل ارقام ۰ تا ۹ می‌باشد و هر مقداری را می‌توان با ترکیبی از ارقام ۰ تا ۹ نمایش داد.
- این سیستم عددی، یک سیستم positional value است. یعنی ارزش هر رقم، بستگی به موقعیت قرارگیری آن رقم دارد.

| | | | | | |
|--------|--------|--------|--------|--------|--------|
| 10^5 | 10^4 | 10^3 | 10^2 | 10^1 | 10^0 |
|--------|--------|--------|--------|--------|--------|

■ مثال

734 : value of 7 : 700 یا 7×100 یا 7×10^2

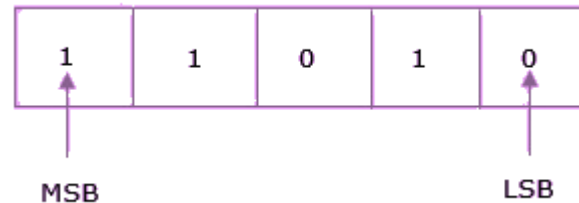
971 : value of 7 : 70 یا 7×10 یا 7×10^1

207 : value of 7 : 7 یا 7×1 یا 7×10^0

- سیستم اعداد باینری، بر مبنای ۲ بوده و شامل ارقام 0 و 1 می باشد و هر مقداری را می توان با ترکیبی از ارقام 0 و 1 نمایش داد.
- به هر رقم باینری یک بیت (bit) گفته می شود.

LSB – Least Significant Bit : سمت راست ترین رقم – بیت با کمترین ارزش عددی

MSB – Most Significant Bit : سمت چپ ترین رقم – بیت با بیشترین ارزش عددی



■ این سیستم عددی، یک سیستم positional value است. یعنی ارزش هر رقم، بستگی به موقعیت قرارگیری آن رقم دارد.

| | | | | | |
|-------|-------|-------|-------|-------|-------|
| 2^5 | 2^4 | 2^3 | 2^2 | 2^1 | 2^0 |
|-------|-------|-------|-------|-------|-------|

■ معادل دسیمال یک عدد باینری عبارت است از مجموع حاصل ضرب هر رقم در ارزش موقعیتی آن رقم.

■ مثال

$$\begin{aligned}11010_2 &= 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\&= 16 + 8 + 0 + 2 + 0 \\&= 26_{10}\end{aligned}$$

- سیستم اعداد اکتال، بر مبنای ۸ بوده و شامل ارقام ۰ تا ۷ می باشد و هر مقداری را می توان با ترکیبی از ارقام ۰ تا ۷ نمایش داد.
- این سیستم عددی نیز یک سیستم positional value است. یعنی ارزش هر رقم، بستگی به موقعیت قرارگیری آن رقم دارد.

| | | | | | |
|-------|-------|-------|-------|-------|-------|
| 8^5 | 8^4 | 8^3 | 8^2 | 8^1 | 8^0 |
|-------|-------|-------|-------|-------|-------|

$$\begin{aligned}
 726_8 &= 7 \times 8^2 + 2 \times 8^1 + 6 \times 8^0 \\
 &= 448 + 16 + 6 \\
 &= 470_{10}
 \end{aligned}$$

■ مثال

■ سیستم اعداد هگز، بر مبنای ۱۶ بوده و شامل ارقام 0 تا 9 و حروف A تا F (معادل 10 تا 15) می باشد.

■ این سیستم عددی نیز یک سیستم positional value است. یعنی ارزش هر رقم، بستگی به موقعیت قرارگیری آن رقم دارد.

| | | | | | |
|--------|--------|--------|--------|--------|--------|
| 16^5 | 16^4 | 16^3 | 16^2 | 16^1 | 16^0 |
|--------|--------|--------|--------|--------|--------|

$$27FB_{16} = 2 \times 16^3 + 7 \times 16^2 + 15 \times 16^1 + 11 \times 16^0$$

$$= 8192 + 1792 + 240 + 11$$

$$= 10235_{10}$$

■ مثال

■ برای **تبدیل مقادیر** در هر کدام از سیستم‌های عددی باینری، اکتال و هگز به **دسیمال** کافی است حاصل ضرب ارقام آن در ارزش مکانی‌شان را با هم جمع کنیم تا به معدل دسیمال آن برسیم.

■ مثال

| | | | | | |
|-------|-------|-------|-------|-------|-------|
| 2^5 | 2^4 | 2^3 | 2^2 | 2^1 | 2^0 |
|-------|-------|-------|-------|-------|-------|

$$\begin{aligned} 11010_2 &= 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\ &= 16 + 8 + 0 + 2 + 0 \\ &= 26_{10} \end{aligned}$$

| | | | | | |
|-------|-------|-------|-------|-------|-------|
| 8^5 | 8^4 | 8^3 | 8^2 | 8^1 | 8^0 |
|-------|-------|-------|-------|-------|-------|

$$\begin{aligned} 726_8 &= 7 \times 8^2 + 2 \times 8^1 + 6 \times 8^0 \\ &= 448 + 16 + 6 \\ &= 470_{10} \end{aligned}$$

| | | | | | |
|--------|--------|--------|--------|--------|--------|
| 16^5 | 16^4 | 16^3 | 16^2 | 16^1 | 16^0 |
|--------|--------|--------|--------|--------|--------|

$$\begin{aligned} 27FB_{16} &= 2 \times 16^3 + 7 \times 16^2 + 15 \times 16^1 + 11 \times 16^0 \\ &= 8192 + 1792 + 240 + 11 \\ &= 10235_{10} \end{aligned}$$

تبدیل دسیمال به باینری

- برای تبدیل دسیمال به باینری، تقسیمات متوالی بر ۲ همراه با ثبت و نگهداری باقیمانده‌ها انجام می‌شود تا وقتی که به خارج قسمت صفر برسیم.
- سپس آخرین باقیمانده را در LSB و به همین ترتیب ادامه می‌دهیم و اولین باقیمانده نیز در MSB قرار می‌گیرد.

| | | Remainder | <div style="display: flex; align-items: center; justify-content: center;"> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">MSB</div> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">↑</div> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">LSB</div> </div> |
|---|----|-----------|--|
| 2 | 43 | | |
| 2 | 21 | 1 | |
| 2 | 10 | 1 | |
| 2 | 5 | 0 | |
| 2 | 2 | 1 | |
| 2 | 1 | 0 | |
| | 0 | 1 | |

$$43_{10} = 101011_2$$

■ مثال

تبدیل دسیمال به اکتال

- برای تبدیل دسیمال به اکتال، تقسیمات متوالی بر ۸ همراه با ثبت و نگهداری باقیمانده‌ها انجام می‌شود تا وقتی که به خارج قسمت صفر برسیم.
- سپس آخرین باقیمانده را در LSB و به همین ترتیب ادامه می‌دهیم و اولین باقیمانده نیز در MSB قرار می‌گیرد.

■ مثال

$$473_{10} = 731_8$$

| Remainder | | |
|-----------|-----|---|
| 8 | 473 | |
| 8 | 59 | 1 |
| 8 | 7 | 3 |
| | 0 | 7 |

MSD
 ↑
 LSD

تبدیل دسیمال به هگز

- برای تبدیل دسیمال به اکتال، تقسیمات متوالی بر ۱۶ همراه با ثبت و نگهداری باقیمانده‌ها انجام می‌شود تا وقتی که به خارج قسمت صفر برسیم.
- سپس آخرین باقیمانده را در LSB و به همین ترتیب ادامه می‌دهیم و اولین باقیمانده نیز در MSB قرار می‌گیرد.

■ مثال

$$423_{10} = 1A7_{16}$$

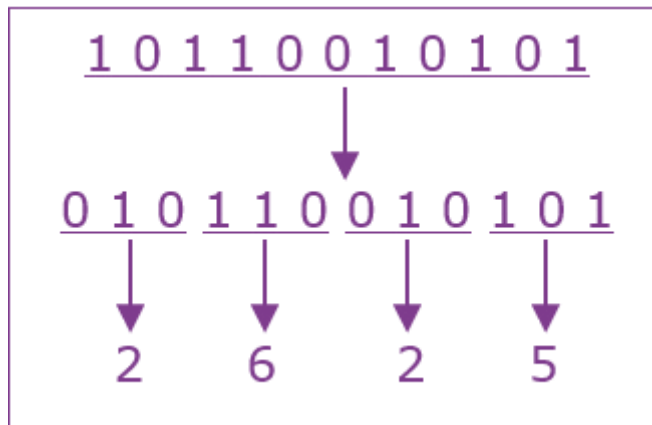
| | | Remainder |
|----|-----|-----------|
| 16 | 423 | |
| 16 | 26 | 7 |
| 16 | 1 | A |
| | 0 | 1 |

تبدیل باینری به اکتال

- برای تبدیل باینری به اکتال، از LSB شروع می کنیم و ارقام عدد را به صورت ۳ رقمی جدا می کنیم.
- اگر گروه آخر تعداد ارقامش کمتر از ۳ بود صفر قرار می دهیم. سپس به جای هر گروه، معادل اکتال آن را می نویسیم.

■ مثال

$$1011001010_{12} = 2625_8$$



تبدیل اکتال به باینری

■ برای تبدیل اکتال به باینری، به جای هر رقم اکتال، معادل باینری آنرا به صورت ۳ رقمی می‌نویسیم.

| Octal Digit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|-------------------|-----|-----|-----|-----|-----|-----|-----|-----|
| Binary Equivalent | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |

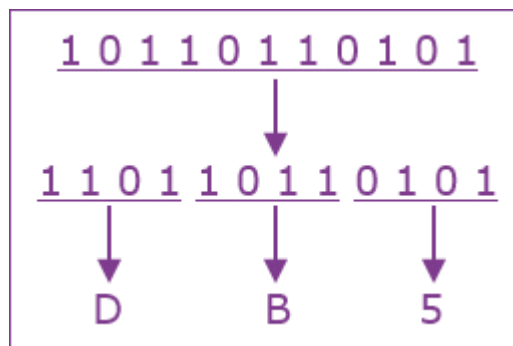
$$54673_8 = 101100110111011_2$$

■ مثال

تبدیل باینری به هگز

- برای تبدیل باینری به هگز، از LSB شروع می کنیم و ارقام عدد را به صورت ۴ رقمی جدا می کنیم.
- اگر گروه آخر تعداد ارقامش کمتر از ۴ بود صفر قرار می دهیم. سپس به جای هر گروه، معادل هگز آن را می نویسیم.

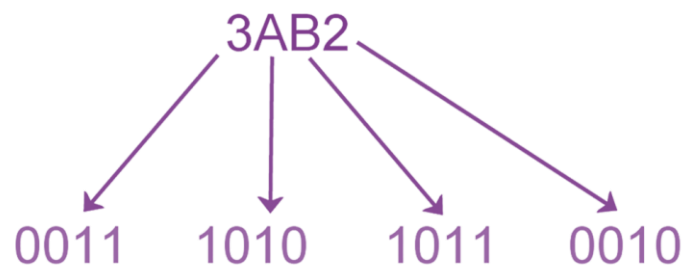
■ مثال



$$10110110101_2 = \text{DB51}_{16}$$

تبدیل هگز به باینری

■ برای تبدیل هگز به باینری، به جای هر رقم هگز، معادل باینری آنرا به صورت ۴ رقمی می نویسیم.



$$3AB2_{16} = 11101010110010_2$$

■ مثال

| HEXADECIMAL | DECIMAL | OCTAL | BINARY |
|-------------|---------|-------|--------|
| 0 | 0 | 0 | 0000 |
| 1 | 1 | 1 | 0001 |
| 2 | 2 | 2 | 0010 |
| 3 | 3 | 3 | 0011 |
| 4 | 4 | 4 | 0100 |
| 5 | 5 | 5 | 0101 |
| 6 | 6 | 6 | 0110 |
| 7 | 7 | 7 | 0111 |
| 8 | 8 | 10 | 1000 |
| 9 | 9 | 11 | 1001 |
| A | 10 | 12 | 1010 |
| B | 11 | 13 | 1011 |
| C | 12 | 14 | 1100 |
| D | 13 | 15 | 1101 |
| E | 14 | 16 | 1110 |
| F | 15 | 17 | 1111 |

- به مجموعه کاراکترهای عددی و غیر عددی مورد استفاده در کامپیوتر، کدهای alphanumeric گفته می‌شود.
- حروف الفبا
- عملگرهای ریاضی
- علائم نگارشی
- علائم و نشانه‌گذاری‌های خاص
- کامپیوتر باید بتواند همه این مقادیر را تشخیص دهد.
- هرچند کامپیوتر فقط قادر به تشخیص 0,1 است ولی اعداد دیگر نیز به کمک تبدیلات سیستم‌های عددی، قابل تبدیل به 0,1 هستند.
- همچنین برای کاراکترهای غیر عددی نیز می‌توان برای هر کاراکتر، یک معادل عددی در نظر گرفت.
- به تکنیک معادل‌سازی کاراکترها با یک مقدار عددی، سیستم کدگذاری اطلاعات (coding) گفته می‌شود.

American Standard Code for Information Interchange (ASCII)

■ سیستم کد ASCII پر کاربردترین سیستم کدینگ اطلاعات است.

■ کد اسکی یک کد ۷ بیتی است که دارای ۱۲۸ کد می باشد.

- 26 upper case letters (A-Z)
- 26 lower case letters (a-z)
- 10 digits (اعداد ۰ تا ۹)
- 7 punctuation marks (علائم نگارشی)
- 20 to 40 special characters (علائم خاص)

ASCII Code - Character to Binary

| | | | |
|-------------|-------------|-------------|-----------------|
| 0 0011 0000 | I 0100 1001 | b 0110 0010 | v 0111 0110 |
| 1 0011 0001 | J 0100 1010 | c 0110 0011 | w 0111 0111 |
| 2 0011 0010 | K 0100 1011 | d 0110 0100 | x 0111 1000 |
| 3 0011 0011 | L 0100 1100 | e 0110 0101 | y 0111 1001 |
| 4 0011 0100 | M 0100 1101 | f 0110 0110 | z 0111 1010 |
| 5 0011 0101 | N 0100 1110 | g 0110 0110 | |
| 6 0011 0110 | O 0100 1111 | h 0110 1000 | : 0011 1010 |
| 7 0011 0110 | P 0101 0000 | i 0110 1001 | ; 0011 1011 |
| 8 0011 1000 | Q 0101 0001 | j 0110 1010 | ? 0011 1111 |
| 9 0011 1001 | R 0101 0010 | k 0110 1011 | · 0010 1110 |
| | S 0101 0011 | l 0110 1100 | ‘ 0010 1111 |
| | T 0101 0100 | m 0110 1101 | ! 0010 0001 |
| A 0100 0001 | U 0101 0101 | n 0110 1110 | ’ 0010 1100 |
| B 0100 0010 | V 0101 0110 | o 0110 1111 | ” 0010 0010 |
| C 0100 0011 | W 0101 0111 | p 0111 0000 | (0010 1000 |
| D 0100 0100 | X 0101 1000 | q 0111 0001 |) 0010 1001 |
| E 0100 0101 | Y 0101 1001 | r 0111 0010 | space 0010 0000 |
| F 0100 0110 | Z 0101 1010 | s 0111 0011 | |
| G 0100 0111 | | t 0111 0100 | |
| H 0100 1000 | a 0110 0001 | u 0111 0101 | |

Indian Script Code for Information Interchange (ISCII)

■ سیستم کدینگ ISCII برای پشتیبانی زبان‌های هندی روی کامپیوتر توسعه داده شد.

- Devanagari
- Tamil
- Bangla
- Gujarati
- Gurmukhi
- Tamil
- Telugu

- یونیکد یک سیستم کدینگ بین‌المللی است که برای پشتیبانی زبان‌های مختلف طراحی شده است.
- قبلاً برای هر زبان، از یک سیستم کدینگ جداگانه استفاده می‌شد که موجب بروز تصادم و ناسازگاری می‌شد.
- یونیکد، بدون توجه به زبان یا پلتفرم و برنامه، برای هر کاراکتر یا سمبول یک مقدار عددی منحصر بفرد در نظر گرفته است.
- یونیکد تا حد زیادی مشابه ASCII است. با این تفاوت که اسکی فقط برای زبان انگلیسی بود ولی یونیکد، تمام زبان‌ها را پشتیبانی می‌کند.

■ کوچکترین واحد اندازه گیری حافظه در کامپیوتر، بیت (bit) است.

■ جدول تبدیل واحدهای حافظه

| | | |
|---------------------------|---|------------|
| 1 byte (B) | = | 8 bits |
| 1 Kilobytes (KB) | = | 1024 bytes |
| 1 Megabyte (MB) | = | 1024 KB |
| 1 Gigabyte (GB) | = | 1024 MB |
| 1 Terabyte (TB) | = | 1024 GB |
| 1 Petabyte (PB) | = | 1024 TB |
| 1 Exabyte (EB) | = | 1024 PB |
| 1 Zettabyte | = | 1024 EB |
| 1 Yottabyte (YB) | = | 1024 ZB |

اصول و مبانی برنامه نویسی



مجید شبیری

کارشناسی ارشد IT، گرایش شبکه
از دانشگاه صنعتی امیرکبیر

