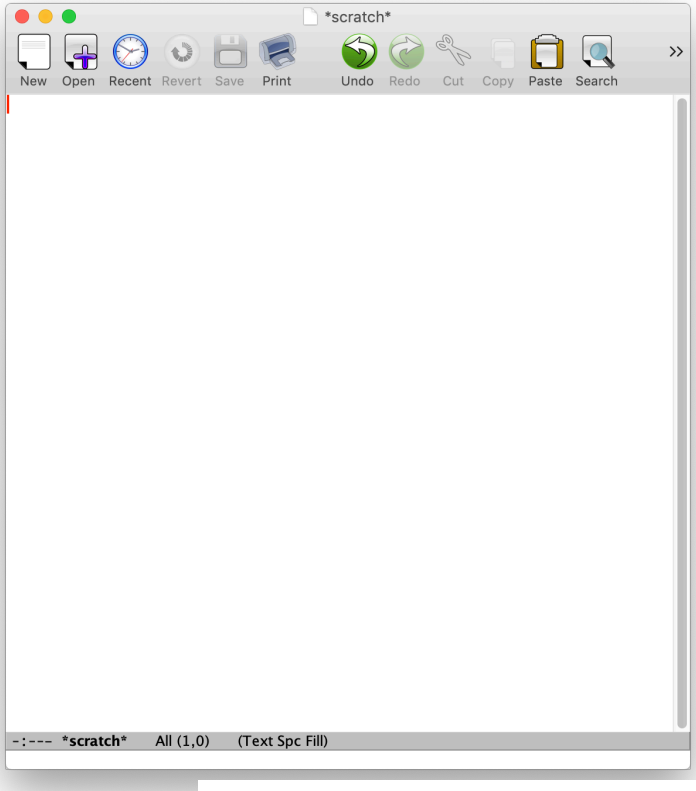
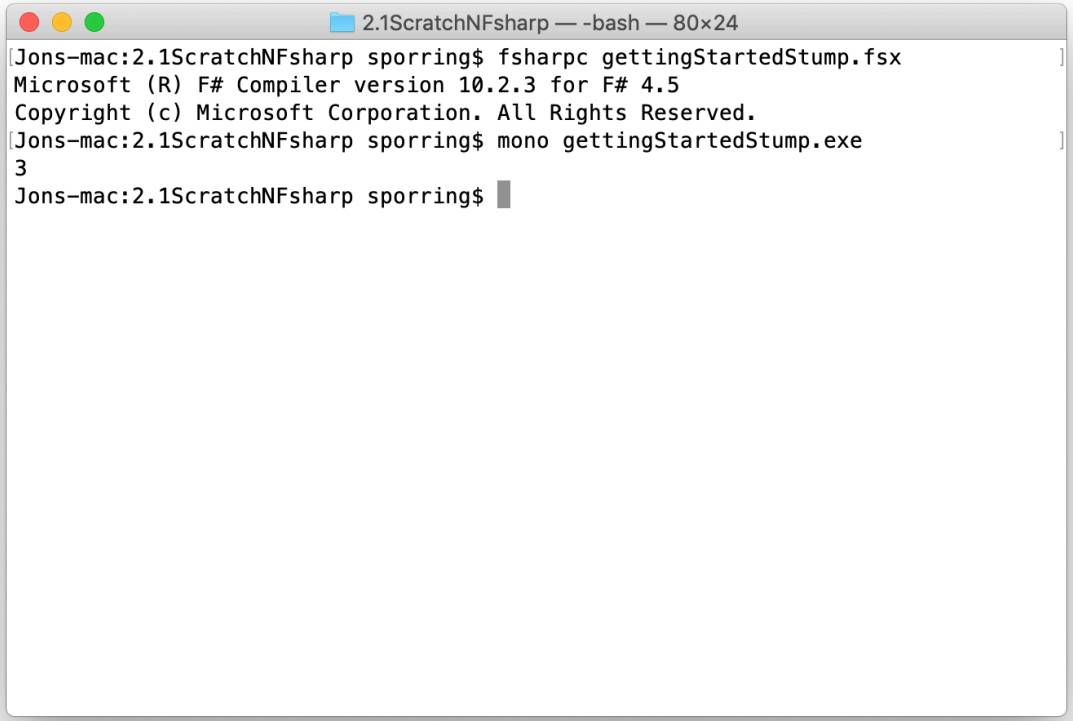
# Programmering og Problemløsning

2.2+3: Tal, typer og operatorer

# F#: Teksteditor, kompilering, kørsel




**Listing 3.1 gettingStartedStump.fsx:**
**A simple demonstration script.**

```
1 let a = 3.0
2 do printfn "%g" a
```

# F#

**Listing 3.1 gettingStartedStump.fsx:**
**A simple demonstration script.**

```
1 let a = 3.0
2 do printfn "%g" a
```

| Command | Time |
|---|---|
| `fsharpi gettingStartedStump.fsx` | 1.88s |
| `fsharpc gettingStartedStump.fsx` | 1.90s |
| `mono gettingStartedStump.exe` | 0.05s |

# Alt på computeren er relateret til binære tal

$$v = \sum_{i=0}^{n} b_i 2^i$$

https://tinyurl.com/ycpxcto5

Et heksal ciffer

Et oktal ciffer

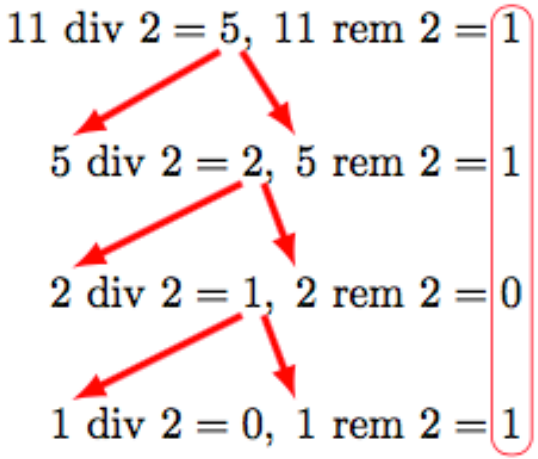
| Dec | Bin | Oct | Hex | Dec | Bin | Oct | Hex |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 32 | 100000 | 40 | 20 |
| 1 | 1 | 1 | 1 | 33 | 100001 | 41 | 21 |
| 2 | 10 | 2 | 2 | 34 | 100010 | 42 | 22 |
| 3 | 11 | 3 | 3 | 35 | 100011 | 43 | 23 |
| 4 | 100 | 4 | 4 | 36 | 100100 | 44 | 24 |
| 5 | 101 | 5 | 5 | 37 | 100101 | 45 | 25 |
| 6 | 110 | 6 | 6 | 38 | 100110 | 46 | 26 |
| 7 | 111 | 7 | 7 | 39 | 100111 | 47 | 27 |
| 8 | 1000 | 10 | 8 | 40 | 101000 | 50 | 28 |
| 9 | 1001 | 11 | 9 | 41 | 101001 | 51 | 29 |
| 10 | 1010 | 12 | a | 42 | 101010 | 52 | 2a |
| 11 | 1011 | 13 | b | 43 | 101011 | 53 | 2b |
| 12 | 1100 | 14 | c | 44 | 101100 | 54 | 2c |
| 13 | 1101 | 15 | d | 45 | 101101 | 55 | 2d |
| 14 | 1110 | 16 | e | 46 | 101110 | 56 | 2e |
| 15 | 1111 | 17 | f | 47 | 101111 | 57 | 2f |
| 16 | 10000 | 20 | 10 | 48 | 110000 | 60 | 30 |
| 17 | 10001 | 21 | 11 | 49 | 110001 | 61 | 31 |
| 18 | 10010 | 22 | 12 | 50 | 110010 | 62 | 32 |
| 19 | 10011 | 23 | 13 | 51 | 110011 | 63 | 33 |
| 20 | 10100 | 24 | 14 | 52 | 110100 | 64 | 34 |
| 21 | 10101 | 25 | 15 | 53 | 110101 | 65 | 35 |
| 22 | 10110 | 26 | 16 | 54 | 110110 | 66 | 36 |
| 23 | 10111 | 27 | 17 | 55 | 110111 | 67 | 37 |
| 24 | 11000 | 30 | 18 | 56 | 111000 | 70 | 38 |
| 25 | 11001 | 31 | 19 | 57 | 111001 | 71 | 39 |
| 26 | 11010 | 32 | 1a | 58 | 111010 | 72 | 3a |
| 27 | 11011 | 33 | 1b | 59 | 111011 | 73 | 3b |
| 28 | 11100 | 34 | 1c | 60 | 111100 | 74 | 3c |
| 29 | 11101 | 35 | 1d | 61 | 111101 | 75 | 3d |
| 30 | 11110 | 36 | 1e | 62 | 111110 | 76 | 3e |
| 31 | 11111 | 37 | 1f | 63 | 111111 | 77 | 3f |

# Dividér med 2 algoritmen

Fra binær til decimal

$$v = \sum_{i=0}^{n} b_i 2^i$$

Fra decimal til binær




https://tinyurl.com/y7s5979a

# Typer definerer formen på legoklodsen

| Type | int | float | char | string | float | float |
|---|---|---|---|---|---|---|
| Tre | 3 | 3.0 | ‘3’ | “3” | 3e0 | 3.0e0 |

| Metatype | Type name | Description |
|---|---|---|
| Boolean | bool | Boolean values true or false |
| Integer | int | Integer values from -2,147,483,648 to 2,147,483,647 |
| | byte | Integer values from 0 to 255 |
| | sbyte | Integer values from -128 to 127 |
| | int8 | Synonymous with sbyte |
| | uint8 | Synonymous with byte |
| | int16 | Integer values from -32768 to 32767 |
| | uint16 | Integer values from 0 to 65535 |
| | int32 | Synonymous with int |
| | uint32 | Integer values from 0 to 4,294,967,295 |
| | int64 | Integer values from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 |
| | uint64 | Integer values from 0 to 18,446,744,073,709,551,615 |
| Real | float | 64-bit IEEE 754 floating point value from −∞ to ∞ |
| | double | Synonymous with float |
| | single | A 32-bit floating point type |
| | float32 | Synonymous with single |
| | decimal | A floating point data type that has at least 28 significant digits |
| Character | char | Unicode character |
| | string | Unicode sequence of characters |
| None | unit | The value () |
| Object | obj | An object |
| Exception | exn | An exception |

| Type | syntax | Examples | Value |
|---|---|---|---|
| int, int32 | <int or hex><br><int or hex>l | 3, 0x3<br>3l, 0x3l | 3 |
| uint32 | <int or hex>u<br><int or hex>ul | 3u<br>3ul | 3 |
| byte, uint8 | <int or hex>uy<br>'<char>'B | 97uy<br>'a'B | 97 |
| byte[] | "<string>"B<br>@"<string>"B | "a\n"B<br>@"a\n"B | [\|97uy; 10uy\|]<br>[\|97uy; 92uy; 110uy\|] |
| sbyte, int8 | <int or hex>y | 3y | 3 |
| int16 | <int or hex>s | 3s | 3 |
| uint16 | <int or hex>us | 3us | 3 |
| int64 | <int or hex>L | 3L | 3 |
| uint64 | <int or hex>UL<br><int or hex>uL | 3UL<br>3uL | 3 |
| float, double | <float><br><hex>LF | 3.0<br>0x013fLF | 3.0<br>9.387247271e-323 |
| single, float32 | <float>F<br><float>f<br><hex>lf | 3.0F<br>3.0f<br>0x013flf | 3.0<br>3.0<br>4.4701421e-43f |
| decimal | <float or int>M<br><float or int>m | 3.0M,3M<br>3.0m,3m | 3.0 |
| string | "<string>"<br>@"<string>"<br>""<string>"" | "a \"quote\".\n"<br>@"a ""quote"".\n"<br>"""a "quote".\n""" | a "quote".<newline><br>a "quote".\n.<br>a "quote".\n |

# ASCII, Latin1, UTF8 og Kodesider

| Character | Escape sequence | Description |
|---|---|---|
| BS | \b | Backspace |
| LF | \n | Line feed |
| CR | \r | Carriage return |
| HT | \t | Horizontal tabulation |
| \ | \\ | Backslash |
| " | \" | Quotation mark |
| ’ | \' | Apostrophe |
| BEL | \a | Bell |
| FF | \f | Form feed |
| VT | \v | Vertical tabulation |
| | \uXXXX, \UXXXXXXXX, \DDD | Unicode character |

| x0+0x | 00 | 10 | 20 | 30 | 40 | 50 | 60 | 70 |
|---|---|---|---|---|---|---|---|---|
| 00 | NUL | DLE | SP | 0 | @ | P | ‘ | p |
| 01 | SOH | DC1 | ! | 1 | A | Q | a | q |
| 02 | STX | DC2 | " | 2 | B | R | b | r |
| 03 | ETX | DC3 | # | 3 | C | S | c | s |
| 04 | EOT | DC4 | $ | 4 | D | T | d | t |
| 05 | ENQ | NAK | % | 5 | E | U | e | u |
| 06 | ACK | SYN | & | 6 | F | V | f | v |
| 07 | BEL | ETB | ’ | 7 | G | W | g | w |
| 08 | BS | CAN | ( | 8 | H | X | h | x |
| 09 | HT | EM | ) | 9 | I | Y | i | y |
| 0A | LF | SUB | * | : | J | Z | j | z |
| 0B | VT | ESC | + | ; | K | [ | k | { |
| 0C | FF | FS | , | < | L | \ | l | \| |
| 0D | CR | GS | – | = | M | ] | m | } |
| 0E | SO | RS | . | > | N | ^ | n | ~ |
| 0F | SI | US | / | ? | O | _ | o | DEL |

Table C.1: ASCII

| x0+0x | 80 | 90 | A0 | B0 | C0 | D0 | E0 | F0 |
|---|---|---|---|---|---|---|---|---|
| 00 | | | NBSP | ° | À | Ð | à | ð |
| 01 | | | ¡ | ± | Á | Ñ | á | ñ |
| 02 | | | ¢ | ² | Â | Ò | â | ò |
| 03 | | | £ | ³ | Ã | Ó | ã | ó |
| 04 | | | ¤ | ´ | Ä | Ô | ä | ô |
| 05 | | | ¥ | µ | Å | Õ | å | õ |
| 06 | | | ¦ | ¶ | Æ | Ö | æ | ö |
| 07 | | | § | · | Ç | × | ç | ÷ |
| 08 | | | ¨ | ¸ | È | Ø | è | ø |
| 09 | | | © | ¹ | É | Ù | é | ù |
| 0a | | | ª | º | Ê | Ú | ê | ú |
| 0b | | | « | » | Ë | Û | ë | û |
| 0c | | | ¬ | ¼ | Ì | Ü | ì | ü |
| 0d | | | SHY | ½ | Í | Ý | í | ý |
| 0e | | | ® | ¾ | Î | Þ | î | þ |
| 0f | | | ¯ | ¿ | Ï | ß | ï | ÿ |

# Operatorer og præcedens

| Operatorer og typer |
| --- |
| 3 + 4 |
| 3.0 + 4.0 |
| ~~3 + 4.0~~ |
| 5 / 2 |
| 5 % 2 |
| 2 * (5 / 2) + 5 % 2 |
| 2.0 ** 3.0 |
| pown 2 3 |
| "hej " + "med " + "dig" |

# Præcedens og association

| Operator | Associativity | Description |
|---|---|---|
| `+<expr>`, `-<expr>`, `~~~<expr>` | Left | Unary identity, negation, and bitwise negation operator |
| `f <expr>` | Left | Function application |
| `<expr> ** <expr>` | Right | Exponent |
| `<expr> * <expr>`, `<expr> / <expr>`, `<expr> % <expr>` | Left | Multiplication, division and remainder |
| `<expr> + <expr>`, `<expr> - <expr>` | Left | Addition and subtraction binary operators |
| `<expr> ^^^ <expr>` | Right | bitwise exclusive or |
| `<expr> < <expr>`, `<expr> <= <expr>`, `<expr> > <expr>`, `<expr> >= <expr>`, `<expr> = <expr>`, `<expr> <> <expr>`, `<expr> <<< <expr>`, `<expr> >>> <expr>`, `<expr> &&& <expr>`, `<expr> \|\|\| <expr>` , | Left | Comparison operators, bitwise shift, and bitwise 'and' and 'or'. |
| `<expr> && <expr>` | Left | Boolean and |
| `<expr> \|\| <expr>` | Left | Boolean or |

# Operatorer og præcedens

| Operatorer og typer | Præcendens og association | Typecasting | Unære operatorer |
|---|---|---|---|
| 3 + 4 | exp 0.0 | float 3 | 2 - 3 |
| 3.0 + 4.0 | exp 1.0 | int 3.2 | -3 |
| 3 + 4.0 | exp 0.0 + 1.0 | int 3.6 | char (int 'c' + -int 'a' + int 'A') |
| 5 / 2 | 2.0 **(3.0 ** 4.0) | int (3.2 + 0.5) = 3 | char (int 'c' -int 'a' + int 'A') |
| 5 % 2 | (2.0 / 3.0)/ 4.0 | int (3.6 + 0.5) = 4 | |
| 2 * (5 / 2) + 5 % 2 | | int 'a' | |
| 2.0 ** 3.0 | | int 'A' | |
| pown 2 3 | | char 65 | |
| "hej " + "med " + "dig" | | char (int 'c' - int 'a' + int 'A') | |