PFI lecture ASCII character set

- bit
  - A unit of information storage. It stores either 0 or 1.
- byte
  - 8 bits. It has 256 values: 00000000 to 1111111, which are all possible combinations.
- K bit has 2^k many combinations.
  - The relation is derived by multiplication principle and inductive reasoning
- char type has one byte or 8 bits.
  - How do we use the 256 combinations to represent a potential 256 characters?
  - ASCII(American Standard Code for Information Interchange)

PFI binary patterns and integers

- 1 bit patterns
  - 0 is integer 0
  - 1 is integer 1
- 2 bit patterns
  - 00 is integer 0
  - 01 is integer 1
  - 10 is integer 2
  - 11 is integer 3
- 3 bit patterns
  - 000(0), 001(1), 010(2), 011(3), 100(4), 101(5), 110(6), 111(7)
- 4 bit patterns
  - 0000(0), 0001(1), 0010(2), 0011(3), 0100(4), 0101(5), 0110(6), 0111(7) 1000(8), 1001(9), 1010(10), 1011(11), 1100(12), 1101(13), 1110(14), 1111(15)

PFI lecture ASCII character set

- The standard ASCII only uses half of the available combination. The first bit is always 0.
- Examples of ASCII
  - Null character, '\0', pattern 00000000 or integer 0
  - Newline, '\n', integer 10
  - Space, ' ', integer 32
  - '0', 48; '1', 49; '2',50;…
  - 'A', 65; 'B',66;…
  - 'a', 97; 'b', 98;…
- A complete ASCII table is available in the internet.
- In C++, **char** type uses ASCII code for the character representation, called internal representation.
- This internal representation is used in **implicit conversion** from char to int. So 'a'*10 gives 970. 'a' > 'A' is true.

PFI lecture ASCII character set

- Examples of logical expressions for char type.
- char c;
- A logical expression for the value of c being a digit:
  - ‘0’ <= c && ‘9’ >= c
- A logical expression for the value of c being a lower case letter
  - ‘a’ <= c && ‘z’ >= c
- The value of c being alphanumeric (digits or letters)
  - ‘0’ <= c && ‘9’ >= c || ‘a’ <= c && ‘z’ >= c || ‘A’ <= c && ‘Z’ >= c
- Convert the lower case letter in variable c to upper case
  - c = c + ‘A’-’a’; // why does this work?

PFI lecture cstring

- In C++ or C, a cstring is an array of char ending with the null character ('\0').
- In fact the type of string literal, such as "Hi", is cstring type.
- Example cstrings:
  - char str[20] = {'H', 'i', '\0'};
  - char greeting[20] = "Hello, World!";
- In using arrays, we are concerned with 3 things: the name of the array, its total size (or capacity), and the number of elements actually have values or data.
- For cstring, even though it is an array, only the name is of concern for cstring functions. Programmers need to be aware of the capacity (array overflow issues)

PFI lecture cstring

- Comparisons of strings (cstrng or string) are based on the ASCII characters in the string.
- Examples:
  - "ABC" is less than "a", even though ABC length is longer
  - "ABB" is less than "ABC"
  - "123456789" is less than "14"
- In C++, string type is built from cstring.
- Many **functions** are available for cstring.