Programmation 1

TD n°1

Amrita Suresh [*]

15/09/2020

Exercise 1 : Language discovery

For each of the following program or fragment of program, please indicate : (a) What the fragment does (b) Is it written using the imperative or functional paradigm (c) In which language the fragment is written.

1.
```
PROGRAM HELLO
WRITE(6,*) 'HELLO WORLD'
STOP
END
```

2.
```
    PROGRAM FACT
    J=1
    DO 1 I=1,10
    J=J*I
1   CONTINUE
    WRITE(6,2) J
2   FORMAT(I8)
    STOP
    END
```

3.
```
001  IDENTIFICATION DIVISION.
002  PROGRAM-ID.  'HELLO'.
003  ENVIRONMENT DIVISION.
004  CONFIGURATION SECTION.
005  SOURCE-COMPUTER.  IBM-360.
006  OBJECT-COMPUTER.  IBM-360.
0065 SPECIAL-NAMES.
0066     CONSOLE IS CNSL.
007  DATA DIVISION.
008  WORKING-STORAGE SECTION.
009  77  HELLO-CONST  PIC X(12) VALUE 'HELLO,WORLD'.
075  PROCEDURE DIVISION.
090  000-DISPLAY.
100      DISPLAY HELLO-CONST UPON CNSL.
110      STOP RUN.
```

4.
```
(defun fact (n)
   (do* ((i 1 (+ i 1)) (j 1 (* j i)))
        ((>= i n) j)))
```

5. $\square \leftarrow */\iota 10$

6.
```
def factorial(n):
    result = 1
```

[*] Majority of the material has been gathered with help from the TAs of this course over the past few years.

```
    for i in range(1, n+1):
        result *= i
    return result
```

7.
```
int fact (int n)
{
  int i, j;

  j = 1;
  for (i=1; i<=n; i++)
     j *= i;
  return j;
}
```

8.
```
fact :: Int -> Int
fact 1 = 1
fact n = n * fact (n-1)
```

9.
```
let rec fact n =
     if n==1
        then 1
     else n * fact (n-1);;
```

10.
```
fact(1, 1).
fact(N, M) :- N > 1, fact (N-1, M1), M=M1*N.
```

11.
```
counter=$1
factorial=1
while [ $counter -gt 0 ]
do
   factorial=$(( $factorial * $counter ))
   counter=$(( $counter - 1 ))
done
echo $factorial
```

12.
```
/factorial {
  dup 1 eq {}{
  dup 1 sub factorial mul
  } ifelse
} def
```

13.
```
function fac(n){
    return(n<2)?1:fac(n-1)*n;
}
```

Exercise 2 : Representation of numbers

1. How many values can a 1 bit integer take ? What about 3 bits ? What about n bits ?
2. You're building a fence 100 feet long, with posts every 10 feet. How many posts do you need ?

Unsigned numbers

The sequence $\vec{a} \triangleq a_{n-1} \cdots a_0$ of digits is interpreted as

$$[\![\vec{a}]\!]_u \triangleq \sum_{k=0}^{n-1} a_k 2^k$$

Two's complement. AKA signed numbers

The sequence $\vec{a} \triangleq a_{n-1} \cdots a_0$ of digits is interpreted as

$$\llbracket \vec{a} \rrbracket_{tc} \triangleq -a_{n-1}2^{n-1} + \sum_{k=0}^{n-2} a_k 2^k$$

3. What values can a natural number represented using n bits take ? What about a signed number ?
4. Compute the following additions on 4 bit unsigned numbers :
   (a) $0010 + 0110$
   (b) $0101 + 1010$
   (c) $1011 + 1101$
   (d) $1010 + 0110$

One's complement

The sequence $\vec{a} \triangleq a_{n-1} \cdots a_0$ of digits is interpreted as

$$\llbracket \vec{a} \rrbracket_{tc} \triangleq \begin{cases} \sum_{k=0}^{n-2} a_k 2^k & \text{if } a_{n-1} = 0 \\ \sum_{k=0}^{n-2} (a_k - 1) 2^k & \text{otherwise} \end{cases}$$

5. How does one write 1 using One's complement ? What about -1 ? How can you negate a number ?
6. What is a huge drawback of this representation ?
7. Using previous examples, build an algorithm to add two numbers in One's complement. (Hint : the question is, how to handle the carry).
8. Why does your algorithm terminate ?
9. What is printed by the Java program below ?

```
byte i = 101, j = 87, k = -101, l = -99;
byte m, n, o;
m = i+j; n = j+k; o = k+l;
System.out.println(m);
System.out.println(n);
System.out.println(o);
```

Exercise 3 : Representation of text

1. Decode the following ASCII string (written using hexadecimal codes)

```
64 6f 6e 27 74 20 70 61 6e 69 63
```

A few Unicode characters

U+000A	LINE FEED (LF)
U+0020	SPACE
U+0021	EXCLAMATION MARK
U+002C	COMMA
U+0030	DIGIT ZERO
U+0041	LATIN CAPITAL LETTER A
U+0061	LATIN SMALL LETTER A

2. What could be the shortcomings of UTF-32 ?

UTF-8 encoding

- — U+0000 à U+007F : 0xxxxxxx
- — U+0080 à U+07FF : 110xxxxx 10xxxxxx
- — U+0800 à U+FFFF : 1110xxxx 10xxxxxx 10xxxxxx
- — U-10000 à U-1FFFFF : 11110xxx 10xxxxxx 10xxxxxx 10xxxxxx

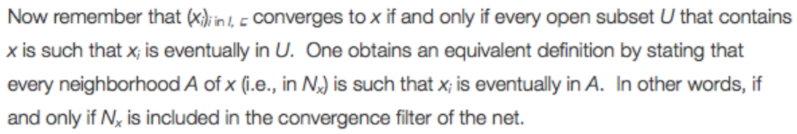
3. Decode the following UTF-8 chain

```
70 65 6E 20 70 69 6E 65 61 70 70 6C 65
```

4. Does UTF-8 have the same shortcomings as UTF-32 ? How and why ?
5. When fetching the following webpage

```
https://projects.lsv.ens-cachan.fr/topology/wp-admin
      /post.php?post=251&action=edit
```

my web browser displays :



Now remember that $(x_i)_{i \text{ in } I,\ \sqsubseteq}$ converges to x if and only if every open subset U that contains x is such that x_i is eventually in U. One obtains an equivalent definition by stating that every neighborhood A of x (i.e., in N_x) is such that x_i is eventually in A. In other words, if and only if N_x is included in the convergence filter of the net.

However, the server `projects.lsv.ens-cachan.fr` sent to my browser the following (extract) of code :

```
Now remember that (<em>x<sub>i</sub></em>) <em><sub>i</sub></em>
<sub> in </sub> <em><sub>I, &sqsubseteq;</sub></em> converges to
<em>x</em> if and only if every open subset <em>U</em> that
contains <em>x</em> is such that <em>x<sub>i</sub></em> is
eventually in <em>U</em>. One obtains an equivalent definition
by stating that every neighborhood <em>A</em> of <em>x</em>
(i.e., in <em>N<sub>x</sub></em>) is such that
<em>x<sub>i</sub></em> is eventually in <em>A</em>.
In other words, if and only if <em>N<sub>x</sub></em>
is included in the convergence filter of the net.
```

How does it compare to Unicode ?

6. Going back to the first example of HTML, the file started with :

```
<?
$EXTRA_HEAD="antispam.html";
$ARG_BODY="onload=\"onLoad()\"";
SETLANG("fr");
STYLEDPTINFO();
HEAD("Conf&eacute;rences de rentr&eacute;e 2015");
ADDTITLE("Conf&eacute;rences de rentr&eacute;e 2015");
MKPAGEDPTINFO();
?>
```

This is not HTML. What language is used ? What does it compute ?