|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| LECTURE | Video títle. | BASIS | Video títle. | ADVANCED |
| Introducció | Introducció | . Presentació, definició, profunditat del temari  . Model de computació: RAM  . Primer programa | Introducció | . Ampliació del temari  . Model de computació: Pointer. |
| Tipus de dades i IO | Tipus de dades I | . Tipus de dades bàsiques. Mides i Operadors.  . Representació en memòria  . Arrays: vectors, taules, multidimensionals | Tipus de dades II | . Tipus de dades ampliades punters. |
| Entrada i sortida. | . Entrada i sortida.  . Formats d’arxiu INI, CSV, dump. |  |  |
| Programació estructurada | Algorísmica Programació estructurada | . Seqüencial.  . Condicional / Alternativa  . Iterativa / fors  . Altres: .) per casos(switch) .) repetitiva |  |  |
| Exemples de programació en pseudocodi | . Seqüencial. 4EX : E1 E2 A1 D1  . Condicional / Alternativa: 4EX : E1 E2 A1 D1  . Iterativa / fors 4EX : E1 E2 A1 D1 |  |  |
| Exemples de programació en C | . Seqüencial. 4EX : E1 E2 A1 D1  . Condicional / Alternativa: 4EX : E1 E2 A1 D1  . Iterativa / fors 4EX : E1 E2 A1 D1 |  |  |
| Programació modular | Programació modular | . Tuples, taules i estructura de dades.  . Accions i funcions.  . Llibreries | Programació modular | . Recursivitat. |
| Exemples de programació en pseudocodi | . Accions, taules i funcions. 4EX : E1 E2 A1 D1  . Tuples i EDA . 4EX : E1 E2 A1 D1 | Exemples de programació en pseudocodi | . Recursivitat. . 4EX : E1 E2 A1 D1 |
| Exemples de programació en C | . Accions, taules i funcions. 4EX : E1 E2 A1 D1  . Tuples i EDA . 4EX : E1 E2 A1 D1 | Exemples de programació en C | . Recursivitat. . 4EX : E1 E2 A1 D1 |
| Disseny de programes | Disseny de programes | . Esquemes algorísmics: Cerca i recorregut. Cerca dico.  . Disseny descendent  . Costos algorismics I | Disseny de programes | . Esquemes algorísmics II: Cerca i ordenació.  . Costos algorísmics II. Recursivitat. |
| Annex. | Annex. | Conceptes històrics (Annex I) | Annex. | . Historia de la informàtica |
|  |  | Conceptes matemàtics (Annex II) |  | . Matemàtica |

**INTRODUCCIÓ**

1.- DEFINICIO

Definicio: word reference Introducció d’un conjunt d’instruccions detallades i codificades a un ordinador perquè executi unes determinades operacions.

INSTRUCCIONS + DETERMINADES OPERACIONS -🡪 LLENGUATGE ALT NIVELL “ANGLES” 🡪 EQUIP/MAQUINES/TERMINAL/ [ELECTRÒNICA] VOLTATGES 🡪COMPILADOR LLENGUATGE HUMÀ / MAQUINA (BINARI -> VOLTATGE)

2.- MODEL DE MÀQUINA

RAM MEM – UPC – IO

Digital Computer Electronics by albert Paul Malvins + jenui97

3.- ALGORISMICA I LLENGUATGES DE PROGRAMACIO

ALG ->PROGRAM

PSEUDOCOG 🡪PROG. LANG

MODEL COMP 🡪 COMPUTADORA

COST-ESPACIAL COST-TEMPORAL

Matemàtiques bàsiques.

4.- CONCEPTES GLOBALS

Constants

Tipus de dades

Codi net

Comentaris

Documentació

5.- [obs] PETITA HISTORIA DE LA PROGRAMACIÓ

- bug / parche

- arxiu

- compilador

- linkar

- swapping

6.- HELLO WORLD EN ALGORÍSMICA.

**(APÈNDIX) MATES BÀSIQUES**

1.- Conversió de dec a binari

2.- Modularització -> parells/senars

3.- Sumes ràpides 🡪 sigma

4.- logaritmes i exponencials

5.- Math libs 🡪 round, abs, trigonometria, upper, lower, max, min resta de doubles

6.- boolea – àlgebra, delta de kronecker.

7.- CA2 numerical.

(APÈNDIX) COMPILACIÓ DE L’IDE

. Warnings

. Error de compilació i el semàntic

. Bones pràctiqus 🡪 documentar-ho

**TIPUS DE DADES / INOUT / LIB**

. Tipus de dades bàsiques

numerical: int, s, long

byte

double/float

logic: bool (bool.h)

string char\*

char str[10]

sentinella \0

sprintf!!!

. Limits dels tipus de dades

. posicionament en la memòria

. datatypes = records

. inout

in scanf, out printf

\n\t

%s %d %lf %d

. punters de memòria

* exercicis a aquest punt. ecoes: manipulació ràpida artimètica
* sperarcio de variables
* concepte de llibreries

PROGRAMACIO ESTRUCTURADA

. Flux

. sequencial

. alternativa/condicional

. iterativa / while / for

evitar break/continue

especials: switch, repeat

(. goto)

A.- evitar A;B;C;

B.- reformateix: if(A){A;T;}else{B;T;} == if(C){A}else{B}T;

C.- Evitar trencar sequencialitat while(c);{a}

D.- evitar continue break 🡪 es pot fer amb un flux

introducció als EA -> while -> ranges.