Plantilla para ejercicios

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Memoria** | |  | **Procesador** | |  | **Programa** | |
| 0 |  |  | A |  |  | 0 |  |
| 1 |  |  | B |  |  | 1 |  |
| 2 |  |  | C |  |  | 2 |  |
| 3 |  |  | PC |  |  | 3 |  |
| 4 |  |  |  |  |  | 4 |  |
| 5 |  |  |  |  |  | 5 |  |
| 6 |  |  |  |  |  | 6 |  |
| 7 |  |  |  |  |  | 7 |  |
| 8 |  |  |  |  |  | 8 |  |
| 9 |  |  |  |  |  | 9 |  |
| 10 |  |  |  |  |  | 10 |  |
| 11 |  |  |  |  |  | 11 |  |
| 12 |  |  |  |  |  | 12 |  |
| 13 |  |  |  |  |  | 13 |  |
| 14 |  |  |  |  |  | 14 |  |
| 15 |  |  |  |  |  | 15 |  |
| 16 |  |  |  |  |  | 16 |  |
| 17 |  |  |  |  |  | 17 |  |
| 18 |  |  |  |  |  | 18 |  |
| 19 |  |  |  |  |  | 19 |  |
| 20 |  |  |  |  |  | 20 |  |
| 21 |  |  |  |  |  | 21 |  |
| 22 |  |  |  |  |  | 22 |  |
| 23 |  |  |  |  |  | 23 |  |
| 24 |  |  |  |  |  | 24 |  |
| 25 |  |  |  |  |  | 25 |  |
| 26 |  |  |  |  |  | 26 |  |
| 27 |  |  |  |  |  | 27 |  |
| 28 |  |  |  |  |  | 28 |  |
| 29 |  |  |  |  |  | 29 |  |
| 30 |  |  |  |  |  | 30 |  |
| 31 |  |  |  |  |  | 31 |  |
| 32 |  |  |  |  |  | 32 |  |

Manual del usuario

|  |  |
| --- | --- |
| 1.- Estructura de una instrucción  Las instrucciones tienen esta forma  CODIGOOPERACION argumento1 argumento2 ...  Ejemplo:  MULTIPLICA @2 3 A  | | | |  | | | +-- Argumento3  | | +---- Argumento2  | +------ Argumento1  +-------------- Código de operación  2.- Argumentos de entrada y de salida  Cada uno de los argumentos de una instrucción  puede ser de entrada o de salida.  En general, los argumentos de entrada se utilizan  para obtener los valores que se van a usar en la  operación.  Los argumentos de salida indican dónde se  almacenarán los resultados de la operación  Ejemplo:    MULTIPLICA @2 3 A  | | |  | | +-- Argumento de entrada  | +---- Argumento de entrada  +------ Argumento de salida | En este caso, se tomará el valor del registro A  y el número 3, se multiplicarán entre sí y el  resultado se almacenará en la celda de memoria @2  3.- Tipos de argumento  Los argumentos pueden ser literales, registros  o celdas de memoria.  3.1) Literal:    Representa un número escrito en decimal.    Ejemplos:    MULTIPLICA @2 3 A  |  |  +---- Literal con valor 3  escrito en decimal      3.2) Registro:  Representan un registro del procesador.    Para referenciarlos tenemos que usar el nombre  del registro.    Ejemplo:    MULTIPLICA @2 3 A  |  +-- Registro A |

|  |  |
| --- | --- |
| 3.3) Celdas de memoria:  Representan una celda de memoria.  Para referenciarlas se antepone @ a la dirección  de la celda.    Ejemplo:    MULTIPLICA @2 3 A  |  |  +------ Celda de memoria  que está en la  dirección 2 |  |

Referencia

|  |  |
| --- | --- |
| ----------------------------------------------------  MUEVE arg1 arg2  Copia en arg1 el valor de arg2.  Sinónimos y abreviaturas: MUE MOVE MOV  ----------------------------------------------------  SUMA arg1 arg2 arg3  Deja en arg1 el resultado de sumar arg2 y arg3.  Sinónimos y abreviaturas: SUM ADD  ----------------------------------------------------  RESTA arg1 arg2 arg3  Deja en arg1 el resultado de restar arg3 de arg2.  Sinónimos y abreviaturas: RES SUBSTRACT SUB  ----------------------------------------------------  MULTIPLICA arg1 arg2 arg3  Deja en arg1 el resultado de multiplicar arg2 con arg3.  Sinónimos y abreviaturas: MUL MULTIPLY  ----------------------------------------------------  DIVIDE arg1 arg2 arg3  Deja en arg1 el resultado de dividir arg2 por arg3.  Sinónimos y abreviaturas: DIV  ---------------------------------------------------- | ----------------------------------------------------  SALTASIMAYOR arg1 arg2 arg3  Salta a la dirección de programa arg1 si arg2 es  mayor que arg3.  Sinónimos y abreviaturas: SMA JUMPIFGREATER JGT  ----------------------------------------------------  SALTASIMAYORIGUAL arg1 arg2 arg3  Salta a la dirección de programa arg1 si arg2 es  mayor o igual que arg3.  Sinónimos y abreviaturas: SMAI JUMPIFGREATEREQUAL JGTE  ----------------------------------------------------  SALTASIMENOR arg1 arg2 arg3  Salta a la dirección de programa arg1 si arg2 es  menor que arg3.  Sinónimos y abreviaturas: SME JUMPIFLESS JLE  ----------------------------------------------------  SALTASIMENORIGUAL arg1 arg2 arg3  Salta a la dirección de programa arg1 si arg2 es  menor o igual que arg3.  Sinónimos y abreviaturas: SMEI JUMPIFLESSEQUAL JLEE  ----------------------------------------------------  SALTASIIGUAL arg1 arg2 arg3  Salta a la dirección de programa arg1 si arg2 es  igual a arg3.  Sinónimos y abreviaturas: SIG JUMPIFEQUAL JEQ  ----------------------------------------------------  SALTASIDIFERENTE arg1 arg2 arg3  Salta a la dirección de programa arg1 si arg2 es  diferente a arg3. |

|  |  |
| --- | --- |
| ----------------------------------------------------  \_LEE\_NUMERO\_ arg1  Macro que almacena en arg1 el valor numérico  correspondiente a una línea de texto introducida  por el usuario. Si no se ha introducido una línea  previamente, pone el procesador en espera hasta que  el usuario la introduzca  ----------------------------------------------------  \_LEE\_CARACTER\_ arg1  Macro que almacena en arg1 el valor numérico  correspondiente a una carácter de texto introducido  por el usuario. Si no se ha introducido un carácter  previamente, pone el procesador en espera hasta que  el usuario lo introduzca  ----------------------------------------------------  \_ESCRIBE\_CARACTER\_ arg1  Macro que muestra como texto el carácter indicado  por arg1  ----------------------------------------------------  \_ESCRIBE\_NUMERO\_ arg1  Macro que muestra como texto el valor numérico  indicado por arg1  ----------------------------------------------------  \_NUMERO\_ALEATORIO\_ arg1 arg2 arg3  Macro que almacena en arg1 un número aleatorio  comprendido entre arg1 y arg2, ambos incluidos | ----------------------------------------------------  \_LIMPIA\_PANTALLA\_  Macro que inicia la memoria de pantalla con el  color negro  ----------------------------------------------------  \_OBTEN\_PIXEL\_ arg1 arg2 arg3  Macro que almacena en arg1 el valor correspondiente  al brillo del pixel con coordenadas (arg2, arg3)  si la profundidad de pantalla es 1 y el valor del canal  rojo del píxel si la profundidad es 3  ----------------------------------------------------  \_PON\_PIXEL\_ arg1 arg2 arg3  Macro que establece el nivel de brillo arg3 para el píxel con  coordenadas (arg1, arg2). El nivel puede ir de 0 a 255  ---------------------------------------------------- |

Apéndice

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tabla de códigos ASCII   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 20 | space | 65 | A | 97 | a | | 33 | ! | 66 | B | 98 | b | | 34 | “ | 67 | C | 99 | c | | 35 | # | 68 | D | 100 | d | | 36 | $ | 69 | E | 101 | e | | 37 | % | 70 | F | 102 | f | | 38 | & | 71 | G | 103 | g | | 39 | ' | 72 | H | 104 | h | | 40 | ( | 73 | I | 105 | i | | 41 | ) | 74 | J | 106 | j | | 42 | \* | 75 | K | 107 | k | | 43 | + | 76 | L | 108 | l | | 44 | , | 77 | M | 109 | m | | 45 | - | 78 | N | 110 | n | | 46 | . | 79 | O | 111 | o | | 47 | / | 80 | P | 112 | p | | 48 | 0 | 81 | Q | 113 | q | | 49 | 1 | 82 | R | 114 | r | | 50 | 2 | 83 | S | 115 | s | | 51 | 3 | 84 | T | 116 | t | | 52 | 4 | 85 | U | 117 | u | | 53 | 5 | 86 | V | 118 | v | | 54 | 6 | 87 | W | 119 | w | | 55 | 7 | 88 | X | 120 | x | | 56 | 8 | 89 | Y | 121 | y | | 57 | 9 | 90 | Z | 122 | z | | 58 | : | 91 | [ | 123 | { | | 59 | ; | 92 | \ | 124 | | | | 60 | < | 93 | ] | 125 | } | | 61 | = | 94 | ^ | 126 | ~ | | 62 | > | 95 | \_ |  |  | | 63 | ? | 96 | @ |  |  | | 64 | @ |  |  |  |  | |  |