|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| HLL | ASSEMBLY CODE | | | | | MACHINE CODE | |
| SECTION | LABEL | MNEMO | OPERANDS | COMMENTS | ADDR | CONT |
|  | .code |  |  |  | ROM: |  |  |
| Input x |  |  | inA |  | A<-Din | 00 |  |
|  |  |  | storeA | X | X<-A | 01 |  |
| Input y |  |  | inA |  | A<-Din | 02 |  |
|  |  |  | storeA | Y | Y<-A | 03 |  |
| if(x>=y){ |  |  | loadA | X | A<-X | 04 |  |
|  |  |  | subA | Y | A<-A-Y | 05 |  |
|  |  |  | Jneg | else | if(A<0)->endif | 06 |  |
|  |  |  | loadA | X |  | 07 |  |
| z=x-y |  |  | subA | Y |  | 08 |  |
|  |  |  | storeA | Z |  | 09 |  |
|  |  |  | Jmp | endif |  | 0A |  |
| }else{ |  | else | loadA | X |  | 0B |  |
| z=x+y |  |  | addA | Y |  | 0C |  |
| } |  |  | storeA | Z |  |  |  |
| While(y>=0){ |  | endif | loadA | Y |  |  |  |
|  |  |  | Jneg | endwhile |  |  |  |
|  |  |  | loadA | X |  |  |  |
|  |  |  | addA | Z |  |  |  |
|  |  |  | storeA | X |  |  |  |
|  |  |  | loadA | Y |  |  |  |
|  |  |  | subA | uno |  |  |  |
|  |  |  | storeA | Y |  |  |  |
| } |  |  | Jmp | endif |  |  |  |
|  |  | endwhile | loadA | Z |  |  |  |
|  |  |  | outA |  |  |  |  |
|  |  |  | loadA | X |  |  |  |
|  |  |  | outA |  |  |  |  |
|  |  |  | halt |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | .data |  |  |  | RAM: |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

|  |  |
| --- | --- |
| Instruction Labels table | |
| label | ADDR  ROM |
| else | 0B |
| Endif | 0E |
| endwhile | 17 |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| Data Labels table | | |
| variable | ADDR  RAM | CONT |
| X | 00 | ?? |
| Y | 01 | ?? |
| Z | 02 | ?? |
| uno | 03 | 01 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |