|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| HLL | ASSEMBLY CODE | | | | | MACHINE CODE | |
| SECTION | LABEL | MNEMO | OPERANDS | COMMENTS | ADDR | CONT |
|  | .code |  |  |  | ROM: |  |  |
|  |  |  |  |  |  |  |  |
| Input N |  |  | inA |  | A<-Din |  |  |
|  |  |  | storeA | N | N<-A |  |  |
|  |  |  |  |  |  |  |  |
| outputN |  |  | loadA | N | A<-N |  |  |
|  |  |  | outputA | N | Dout<-A |  |  |
|  |  |  |  |  |  |  |  |
| if(N>=0){ |  |  | loadA | N | A<-N |  |  |
|  |  |  | jneg | endif | if(A<0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
| if(N<=0){ |  |  | loadA | N | A<-N |  |  |
|  |  |  | jpos | endif | if(A>0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
| if(X>=Y){ |  |  | loadA | X | A<-X |  |  |
|  |  |  | subA | Y | A<-A-Y |  |  |
|  |  |  | jneg | endif | if(A<0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| if(X<=Y){ |  |  | loadA | X | A<-X |  |  |
|  |  |  | subA | Y | A<-A-Y |  |  |
|  |  |  | jpos | endif | if(A>0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
| if(N==0){ |  |  | loadA | N | A<-N |  |  |
|  |  |  | jnz | endif | if(A!=0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
| if(N!=0){ |  |  | loadA | N | A<-N |  |  |
|  |  |  | jz | endif | if(A==0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
| if(X==Y){ |  |  | loadA | X | A<-X |  |  |
|  |  |  | subA | Y | A<-A-Y |  |  |
|  |  |  | jnz | endif | if(A!=0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
| if(X!=Y){ |  |  | loadA | X | A<-X |  |  |
|  |  |  | subA | Y | A<-A-Y |  |  |
|  |  |  | jz | endif | if(A==0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| if(N>0){ |  |  | loadA | N | A<-N |  |  |
|  |  |  | jneg | endif | if(A<0)->endif |  |  |
|  |  |  | jz | endif | if(A==0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
| if(N<0){ |  |  | loadA | N | A<-N |  |  |
|  |  |  | jpos | endif | if(A>0)->endif |  |  |
|  |  |  | jz | endif | if(A==0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
| if(X>Y){ |  |  | loadA | X | A<-X |  |  |
|  |  |  | subA | Y | A<-A-Y |  |  |
|  |  |  | jneg | endif | if(A<0)->endif |  |  |
|  |  |  | jz | endif | if(A==0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
| if(X<Y){ |  |  | loadA | X | A<-X |  |  |
|  |  |  | subA | Y | A<-A-Y |  |  |
|  |  |  | jpos | endif | if(A>0)->endif |  |  |
|  |  |  | jz | endif | if(A==0)->endif |  |  |
| //do something |  |  | //TODO |  |  |  |  |
| } |  | endif | halt |  | endif |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Prototipo if-else |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| if(condition){ |  |  | loadA | statements |  |  |  |
|  |  |  | jncondition | else |  |  |  |
| //Do something |  |  |  |  |  |  |  |
|  |  |  | jmp | endif |  |  |  |
| }else{ |  | else | instructions |  |  |  |  |
| //Do something |  |  |  |  |  |  |  |
| } |  | endif | halt |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Prototipo while |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| while(condition){ |  | while | loadA | statements |  |  |  |
|  |  |  | jncondition | endwhile |  |  |  |
| //Do something |  |  |  |  |  |  |  |
| } |  |  | jmp | while |  |  |  |
|  |  | endwhile | halt |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Prototipo for |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| for(init;condition;update){ |  |  | init | statements |  |  |  |
|  |  | for | loadA | statements |  |  |  |
|  |  |  | jncondition | endfor |  |  |  |
| //Do something |  |  |  |  |  |  |  |
|  |  |  | update | statements |  |  |  |
| } |  |  | jmp | for |  |  |  |
|  |  | endfor | halt |  |  |  |  |
|  |  |  |  |  |  |  |  |

|  |  |
| --- | --- |
| Instruction Labels table | |
| label | ADDR  ROM |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| Data Labels table | | |
| variable | ADDR  RAM | CONT |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |