

Diagramma a Blocchi (DAB)	Programma C/C++ conv_dec_bin.cpp	Pseudocodifica
<pre> graph TD     INIZIO([INIZIO]) --&gt; pot1[pot ← 1]     pot1 --&gt; scrivi1[/SCRIVI (TIT, ACAPO1, ACAPO, ASK)/]     scrivi1 --&gt; leggi[/LEGGI (num)/]     leggi --&gt; pot2[pot ← pot * 2]     pot2 --&gt; dec1{pot &lt;= num}     dec1 -- V --&gt; pot2     dec1 -- F --&gt; scrivi2[/SCRIVI (ACAPO, PSUP, pot, ACAPO)/]     scrivi2 --&gt; k1[K ← 1]     k1 --&gt; dec2{K &lt;= 26}     dec2 -- V --&gt; scrivi3[/SCRIVI (MENO)/]     scrivi3 --&gt; k2[K ← K + 1]     k2 --&gt; dec2     dec2 -- F --&gt; scrivi4[/SCRIVI (ACAPO2, COD)/]     scrivi4 --&gt; pot3[pot ← pot / 2]     pot3 --&gt; dec3{pot &gt; 0}     dec3 -- F --&gt; fine([FINE])     dec3 -- V --&gt; dec4{num &gt;= pot}     dec4 -- F --&gt; scrivi5[/SCRIVI (0)/]     dec4 -- V --&gt; scrivi6[/SCRIVI (1)/]     scrivi6 --&gt; num1[num ← num - pot]     num1 --&gt; dec3     dec3 --&gt; pot4[pot ← pot / 2]     pot4 --&gt; dec3     </pre>	<pre> #include &lt;iostream&gt; using namespace std; #define TIT "Conversione da base 10 a 2" #define ASK "digita num. intero &gt; 0: " #define PSUP "1^ potenza di 2 &gt; num : " #define COD "codifica in base 2 : " #define MENO '-' #define ACAPO1 '\n' #define ACAPO2 "\n\n" int main(){     int num, k, pot=1;     cout &lt;&lt; TIT &lt;&lt; ACAPO1;     cout &lt;&lt; endl &lt;&lt; ASK; //ACAPO1 è come endl     cin &gt;&gt; num;     do {         pot = pot*2;     } while (pot &lt;= num); //FINE del do...while      cout &lt;&lt; endl &lt;&lt; PSUP &lt;&lt; pot &lt;&lt; endl;      for(k=1; k &lt;= 26 ; k++){         cout &lt;&lt; MENO;     } //FINE del for      cout &lt;&lt; ACAPO2 &lt;&lt; COD;     pot = pot/2;      while (pot &gt; 0){         if (num &gt;= pot){ //se Vero / ALLORA             cout &lt;&lt; 1 ;             num = num - pot;         }         else // se Falso / ALTRIMENTI             cout &lt;&lt; 0 ; //FINE del SE          pot = pot/2;     } //FINE del while }     </pre>	<p>INIZIO</p> <p>pot ← 1</p> <p>SCRIVI (TIT, ACAPO1)</p> <p>SCRIVI (ACAPO, ASK)</p> <p>LEGGI (num)</p> <p><b>ESEGUI</b></p> <p>pot ← pot * 2</p> <p><b>RIPETI MENTRE</b> pot &lt;= num</p> <p>SCRIVI (ACAPO, PSUP, pot, ACAPO)</p> <p><b>PER k DA 1 A 26 PASSO 1</b></p> <p>SCRIVI (MENO)</p> <p><b>RIPETI</b></p> <p>SCRIVI (ACAPO2, COD)</p> <p>pot ← pot / 2</p> <p><b>ESEGUI MENTRE</b> pot &gt; 0</p> <p><b>SE</b> num &gt;= pot</p> <p><b>ALLORA</b></p> <p>SCRIVI (1)</p> <p>num ← num - pot</p> <p><b>ALTRIMENTI</b></p> <p>SCRIVI (0)</p> <p><b>FINE SE</b></p> <p>pot ← pot / 2</p> <p><b>RIPETI</b></p> <p>FINE</p>
<p>Si converte num determinando la prima potenza di 2 superiore a num e procedendo poi per sottrazioni successive</p>	<p><b>OUTPUT →</b></p> <pre> Conversione da base 10 a 2 digita num. intero &gt; 0: 23 1^ potenza di 2 &gt; num : 32 ----- codifica in base 2 : 10111     </pre>	