

Code Optimization-

The process of code optimization involves-

- Eliminating the unwanted code lines
- Rearranging the statements of the code

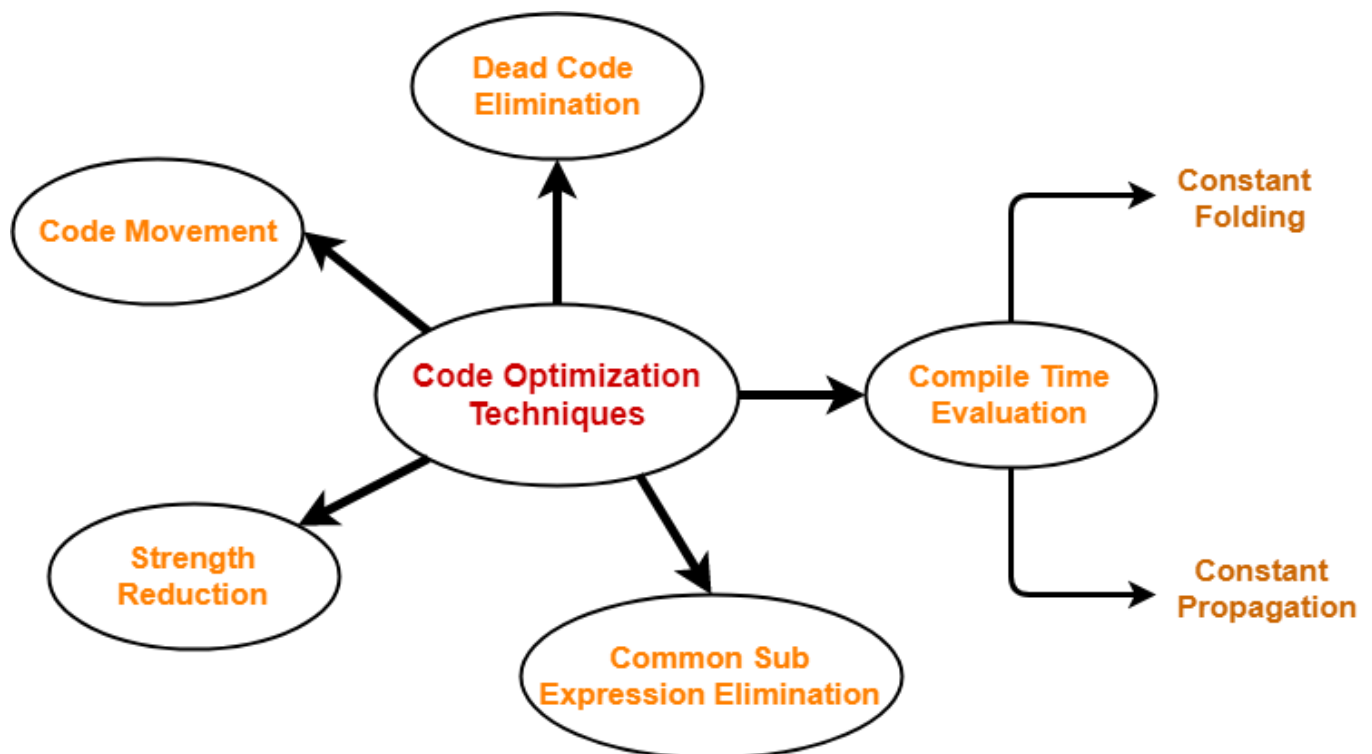
Advantages-

The optimized code has the following advantages-

- Optimized code has faster execution speed.
- Optimized code utilizes the memory efficiently.
- Optimized code gives better performance.

Code Optimization Techniques-

Important code optimization techniques are-



- This technique evaluates the expression $22/7$ at compile time.
- The expression is then replaced with its result 3.14.

- This saves the time at run time.

B) Constant Propagation-

In this technique,

- If some variable has been assigned some constant value, then it replaces that variable with its constant value in the further program during compilation.
- The condition is that the value of variable must not get alter in between.

Example-

pi = 3.14

The expression that has been already computed before and appears again in the code for computation
is called as **Common Sub-Expression**.

Code Before Optimization	Code After Optimization
S1 = 4 x i S2 = a[S1] S3 = 4 x j S4 = 4 x i // Redundant Expression S5 = n S6 = b[S4] + S5	S1 = 4 x i S2 = a[S1] S3 = 4 x j S5 = n S6 = b[S1] + S5

Code Before Optimization	Code After Optimization
for (int j = 0 ; j < n ; j ++) {	x = y + z ; for (int j = 0 ; j < n ; j ++) {

<pre>x = y + z ; a[j] = 6 x j; }</pre>	<pre>{ a[j] = 6 x j; }</pre>
--	------------------------------

Code Before Optimization	Code After Optimization
<pre>i = 0 ; if (i == 1) { a = x + 5 ; }</pre>	<pre>i = 0 ;</pre>

Code Before Optimization	Code After Optimization
<pre>B = A x 2</pre>	<pre>B = A + A</pre>