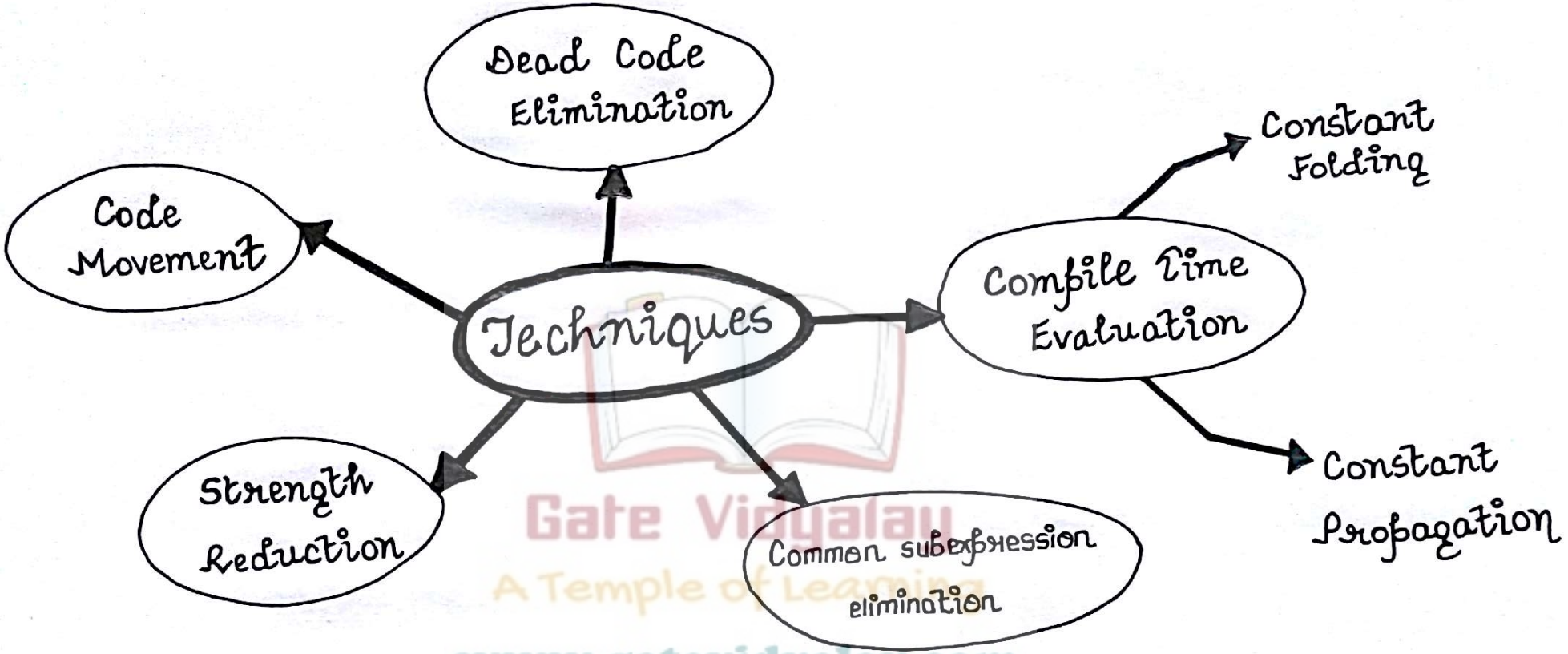
Code optimization

Definition

Code Optimization is a technique which tries to improve the code by eliminating unnecessary code lines and arranging the statements in such a sequence that speed up the program execution without wasting the resources.

Advantages :-

- → Executes faster
- → Efficient memory usage
- → Yields better performance



① Compile Time Evaluation

i) Constant Folding :-

It refers to a technique of evaluating the expressions whose operands are known to be constant at compile time itself.

Example :- length = (2217) * d

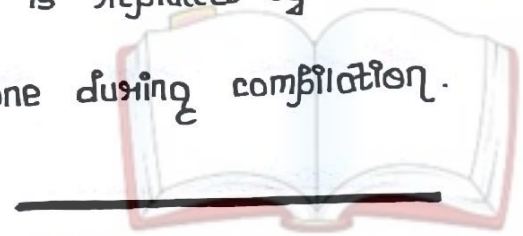
ii) Constant Propagation :-

In constant propagation, if a variable is assigned a constant value, then subsequent use of that variable can be replaced by a constant as long as no intervening assignment has changed the value of the variable.

Example:-

```
pi = 3.14
r = 5
Area = pi * r * r
```

Here, the value of pi is replaced by 3.14 and r by 5, then computation of 3.14 * r * r is done during compilation.



⑪ Common sub-expression elimination :-

The common sub-expression is an expression appearing repeatedly in the code which is computed previously. This technique replaces redundant expression each time it is encountered.

Example :-

Before Optimization

```
T1 = 4*i
T2 = a[T1]
T3 = 4*j
T4 = 4*i
T5 = n
T6 = b[T4] + T5
```

After Optimization

```
T1 = 4*i
T2 = a[T1]
T3 = 4*j
T5 = n
T6 = b[T4] + T5
```

(iii) Code Movement:-

It is a technique of moving a block of code outside a loop if it went have any difference if it is executed outside or inside the loop.

Example:-

Before Optimization

```
y=5
z=6
for (int i=0; i<n; i++)
{
    x = y+z;
    a[i] = 6*i;
}
```

After Optimization

```
y=5
z=6
x = y+z;
for (int i=0; i<n; i++)
{
    a[i] = 6*i;
}
```

(iv) Dead Code Elimination:-

Dead Code Elimination includes eliminating those code statements which are either never executed or unreachable or if executed their output is never used.

Example:-

```
i=0
if (i==1)
{
    a=x+5;
}
```

Before Optimization

```
i=0
```

After Optimization

(v) Strength Reduction:-

It is the replacement of expressions that are expensive with cheaper and simple ones.

Example:-

Before Optimization	After Optimization
B = A * 2	B = A + A