Compiler Optimize Exercise

Given this loop code fragment in an Intermediate file of a Compiler:

```
(1)   :=    #1          Indx
(2)   BGT   Indx  #8    (20)
(3)   -     Indx  #1    i1
(4)   *     i1    #10   i2
(5)   *     #5    DEF   i3
(6)   -     i3    #1    i4
(7)   -     i4    #1    i5
(8)   +     i2    i5    i6
(9)   *     i6    #4    i7
(10)  -     Indx  #1    i8
(11)  *     i8    #10   i9
(12)  *     #5    DEF   i10
(13)  -     i10   #1    i11
(14)  +     i9    i11   i12
(15)  *     i12   #4    i13
(16)  :=    Y[i13]      X[i7]
(17)  +     #1    Indx  i14
(18)  :=    i14         Indx
(19)  JMP               (2)
(20)
```

Indicate the significant code Optimizations.

[50 points]

Some of the possible optimization techniques:

1. Move loop invariant calculations outside the loop
2. Remove duplicate common sub-expressions
3. Reduction in Strength
4. Folding

Not all of the above possible methods are needed .