**X86 code**

```
LVAR
```

Sim := print (Exp)
| name = Exp
| Exp

Exp := Exp + Exp
| Exp - Exp
| - Exp
| Const
| (EXP, EXP)
| name

-> X = 3

```
y = int (x - int (x))
print (x * y)
```

```
x = 42
x = (x + 7 + y)
print (z)
```

```
x = z + 2
```

**X86 - 64** ARCHITECTURE

APPLICATION BINARY INTERFACE (ABI)

**REGISTERS 16 64-BIT**

**MEMORY**

**INSTRUCTIONS**

```
addq, subq, negq, movq
```

```
addq 90ra1, 90rbx
```

```
addq 832, 90rbx
```

```
addq 932, 90rbp
```

```
movq -16 (80ra1), 90rbx
```

```
movq -16 (80ra1), 32 (80rbx)
```

```
add 32
```

```
movs
```

```
8 bytes
```

```
4 bytes
```
main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main:

body of code

main: