Main.ash

```
main {
     //take input
    ---input----
    jmpi _ret; lbl _in = $$
    storeir $input; lbl _ret = $$
    ---input-
    //call div
    call &ret51; lbl loop = $$
    push $input
    push $five
    //output result
    outrn div.$return; lbl ret51 = $$
    //call mod
    call &ret52
push $input
    push $five
    //assign result of mod to input
    loada mod.$return; lbl ret52 = $$
    //call div
    call &ret21
    push $input
    push $two
    //output result
    outrn div.$return; lbl ret21 = $$
    //call mod
    call &ret22
    push $input
    push $two
    //assign result of mod to input
    loada mod.$return; lb1 ret22 = $$
    //call div
    call &ret11
    push $input
    push $one
    //output result
    outrn div.$return; lbl ret11 = $$
    //call mod
    call &ret12
    push $input
    jmp $mod
    //assign result of mod to input
    loada mod.$return; lbl ret12 = $$
    storea $input
```

```
//call div
     call &reta1
     push $input
     push $ten
     loada div.$return; lbl reta1 = $$
     //call div
     call &reta2
     push $input
     push $ten
jmp $div
     loada div.$return; lbl reta2 = $$
     //call div
     call &reta3
     push $input
push $ten
     loada div.$return; lbl reta3 = $$
     storea $five
     //compare results of div
     halt
}
func div {
     loada $a
loadb $b; lbl _sub = $$
subab $r
     loada $r
     loadb null
     jmps $_end
     loada $one
loadb $count
addab $count
     loada $r
     loada $count; lbl _end = $$
     ret
}
```

```
func mod {
    loada $a
    loadb $b
    subab $r; lbl _sub = $$
    loada $r
    jmps $_end
    jmp $_sub
    loada $r; lbl _end = $$
    storea $return
    ret
}
```

Header.ash

```
global {
    const fivet: int = 5000
    const twot: int = 2000
    const onet: int = 1000
    const ten: int = 10

    var five: int = fivet
    var two: int = twot
    var one: int = onet
    var input: int = later
}

func div(a: int, b: int) -> return: int {
    var r: int = later
    const one: int = 1
    var count: int = null
}

func mod(a: int, b: int) -> return: int {
    var r: int = later
}
```

