

Dump of assembler code for function main:

```
0x00000000100000ef0 <+0>:    push    rbp
0x00000000100000ef1 <+1>:    mov     rbp, rsp
0x00000000100000ef4 <+4>:    sub     rsp, 0x20
0x00000000100000ef8 <+8>:    lea     rdi, [rip+0x9e]      #
0x100000f9d
0x00000000100000eff <+15>:   lea     rsi, [rbp-0xa]
0x00000000100000f03 <+19>:   mov     DWORD PTR [rbp-0x4], 0x0
0x00000000100000f0a <+26>:   mov     al, 0x0
0x00000000100000f0c <+28>:   call    0x100000f5c
0x00000000100000f11 <+33>:   lea     rdi, [rbp-0xa]
0x00000000100000f15 <+37>:   mov     DWORD PTR [rbp-0x10], eax
0x00000000100000f18 <+40>:   call    0x100000e40 <_Z5Function2>
0x00000000100000f1d <+45>:   cmp     eax, 0x0
0x00000000100000f20 <+48>:   je      0x100000f3c <main+76>
0x00000000100000f26 <+54>:   lea     rdi, [rip+0x75]      #
0x100000fa2
0x00000000100000f2d <+61>:   mov     al, 0x0
0x00000000100000f2f <+63>:   call    0x100000f56
0x00000000100000f34 <+68>:   mov     DWORD PTR [rbp-0x14], eax
0x00000000100000f37 <+71>:   jmp     0x100000f4d <main+93>
0x00000000100000f3c <+76>:   lea     rdi, [rip+0x69]      #
0x100000fac
0x00000000100000f43 <+83>:   mov     al, 0x0
0x00000000100000f45 <+85>:   call    0x100000f56
0x00000000100000f4a <+90>:   mov     DWORD PTR [rbp-0x18], eax
0x00000000100000f4d <+93>:   mov     eax, DWORD PTR [rbp-0x4]
0x00000000100000f50 <+96>:   add     rsp, 0x20
0x00000000100000f54 <+100>:  pop     rbp
0x00000000100000f55 <+101>:  ret
```

Dump of assembler code for function Function2:

```
0x00000000100000e40 <+0>:    push    rbp
0x00000000100000e41 <+1>:    mov     rbp, rsp
0x00000000100000e44 <+4>:    sub     rsp, 0x20
0x00000000100000e48 <+8>:    mov     QWORD PTR [rbp-0x10], rdi
0x00000000100000e4c <+12>:   mov     rdi, QWORD PTR [rbp-0x10]
0x00000000100000e50 <+16>:   call    0x100000f62
0x00000000100000e55 <+21>:   cmp     rax, 0x5
0x00000000100000e59 <+25>:   je      0x100000e6b <_Z5Function2+43>
0x00000000100000e5f <+31>:   mov     DWORD PTR [rbp-0x4], 0x0
0x00000000100000e66 <+38>:   jmp     0x100000ee6 <_Z5Function2+166>
0x00000000100000e6b <+43>:   mov     rdi, QWORD PTR [rbp-0x10]
0x00000000100000e6f <+47>:   call    0x100000dc0 <_Z13Function3>
0x00000000100000e74 <+52>:   cmp     eax, 0x0
0x00000000100000e77 <+55>:   jne     0x100000e89 <_Z5Function2+73>
0x00000000100000e7d <+61>:   mov     DWORD PTR [rbp-0x4], 0x0
0x00000000100000e84 <+68>:   jmp     0x100000ee6 <_Z5Function2+166>
0x00000000100000e89 <+73>:   mov     rdi, QWORD PTR [rbp-0x10]
0x00000000100000e8d <+77>:   call    0x100000d70 <_Z4Function4Pc>
0x00000000100000e92 <+82>:   lea     rdi, [rip+0xfd]      #
0x100000f96
0x00000000100000e99 <+89>:   mov     ecx, 0xa
0x00000000100000e9e <+94>:   mov     DWORD PTR [rbp-0x14], eax
```

```

0x00000000100000ea1 <+97>: mov     eax,DWORD PTR [rbp-0x14]
0x00000000100000ea4 <+100>: cdq
0x00000000100000ea5 <+101>: idiv   ecx
0x00000000100000ea7 <+103>: mov     esi,edx
0x00000000100000ea9 <+105>: mov     al,0x0
0x00000000100000eab <+107>: call    0x100000f56
0x00000000100000eb0 <+112>: mov     ecx,0xa
0x00000000100000eb5 <+117>: mov     edx,DWORD PTR [rbp-0x14]
0x00000000100000eb8 <+120>: mov     DWORD PTR [rbp-0x18],eax
0x00000000100000ebb <+123>: mov     eax,edx
0x00000000100000ebd <+125>: cdq
0x00000000100000ebe <+126>: idiv   ecx
0x00000000100000ec0 <+128>: mov     rdi,QWORD PTR [rbp-0x10]
0x00000000100000ec4 <+132>: movsx   ecx,BYTE PTR [rdi+0x4]
0x00000000100000ec8 <+136>: sub     ecx,0x30
0x00000000100000ecb <+139>: cmp     edx,ecx
0x00000000100000ecd <+141>: je      0x100000edf <_Z5Function2+159>
0x00000000100000ed3 <+147>: mov     DWORD PTR [rbp-0x4],0x0
0x00000000100000eda <+154>: jmp     0x100000ee6 <_Z5Function2+166>
0x00000000100000edf <+159>: mov     DWORD PTR [rbp-0x4],0x1
0x00000000100000ee6 <+166>: mov     eax,DWORD PTR [rbp-0x4]
0x00000000100000ee9 <+169>: add     rsp,0x20
0x00000000100000eed <+173>: pop     rbp
0x00000000100000eee <+174>: ret
0x00000000100000eef <+175>: nop

```

Dump of assembler code for function Function3:

```

0x00000000100000dc0 <+0>: push    rbp
0x00000000100000dc1 <+1>: mov     rbp,rsp
0x00000000100000dc4 <+4>: mov     QWORD PTR [rbp-0x10],rdi
0x00000000100000dc8 <+8>: mov     DWORD PTR [rbp-0x14],0x0
0x00000000100000dcf <+15>: cmp     DWORD PTR [rbp-0x14],0x5
0x00000000100000dd3 <+19>: jge     0x100000e26 <_Z13Function3+102>
0x00000000100000dd9 <+25>: movsxd  rax,DWORD PTR [rbp-0x14]
0x00000000100000ddd <+29>: mov     rcx,QWORD PTR [rbp-0x10]
0x00000000100000de1 <+33>: movsx   edx,BYTE PTR [rcx+rax*1]
0x00000000100000de5 <+37>: cmp     edx,0x30
0x00000000100000de8 <+40>: jl      0x100000e07 <_Z13Function3+71>
0x00000000100000dee <+46>: mov     eax,0x39
0x00000000100000df3 <+51>: movsxd  rcx,DWORD PTR [rbp-0x14]
0x00000000100000df7 <+55>: mov     rdx,QWORD PTR [rbp-0x10]
0x00000000100000dfb <+59>: movsx   esi,BYTE PTR [rdx+rcx*1]
0x00000000100000dff <+63>: cmp     eax,esi
0x00000000100000e01 <+65>: jge     0x100000e13 <_Z13Function3+83>
0x00000000100000e07 <+71>: mov     DWORD PTR [rbp-0x4],0x0
0x00000000100000e0e <+78>: jmp     0x100000e2d <_Z13Function3+109>
0x00000000100000e13 <+83>: jmp     0x100000e18 <_Z13Function3+88>
0x00000000100000e18 <+88>: mov     eax,DWORD PTR [rbp-0x14]
0x00000000100000e1b <+91>: add     eax,0x1
0x00000000100000e1e <+94>: mov     DWORD PTR [rbp-0x14],eax
0x00000000100000e21 <+97>: jmp     0x100000dcf <_Z13Function3+15>
0x00000000100000e26 <+102>: mov     DWORD PTR [rbp-0x4],0x1
0x00000000100000e2d <+109>: mov     eax,DWORD PTR [rbp-0x4]
0x00000000100000e30 <+112>: pop     rbp

```

```
0x00000000100000e31 <+113>: ret
0x00000000100000e32 <+114>: data16 data16 data16 data16 nop WORD
PTR cs:[rax+rax*1+0x0]
```

Dump of assembler code for function Function4:

```
0x00000000100000d70 <+0>: push    rbp
0x00000000100000d71 <+1>: mov     rbp, rsp
0x00000000100000d74 <+4>: mov     QWORD PTR [rbp-0x8], rdi
0x00000000100000d78 <+8>: mov     DWORD PTR [rbp-0xc], 0x0
0x00000000100000d7f <+15>: mov     DWORD PTR [rbp-0x10], 0x0
0x00000000100000d86 <+22>: cmp     DWORD PTR [rbp-0x10], 0x4
0x00000000100000d8a <+26>: jge     0x100000db9 <_Z4Function4Pc+73>
0x00000000100000d90 <+32>: movsxd  rax, DWORD PTR [rbp-0x10]
0x00000000100000d94 <+36>: mov     rcx, QWORD PTR [rbp-0x8]
0x00000000100000d98 <+40>: movsx   edx, BYTE PTR [rcx+rax*1]
0x00000000100000d9c <+44>: sub     edx, 0x30
0x00000000100000d9f <+47>: imul    edx, edx, 0xffff
0x00000000100000da5 <+53>: add     edx, DWORD PTR [rbp-0xc]
0x00000000100000da8 <+56>: mov     DWORD PTR [rbp-0xc], edx
0x00000000100000dab <+59>: mov     eax, DWORD PTR [rbp-0x10]
0x00000000100000dae <+62>: add     eax, 0x1
0x00000000100000db1 <+65>: mov     DWORD PTR [rbp-0x10], eax
0x00000000100000db4 <+68>: jmp     0x100000d86 <_Z4Function4Pc+22>
0x00000000100000db9 <+73>: mov     eax, DWORD PTR [rbp-0xc]
0x00000000100000dbc <+76>: pop     rbp
0x00000000100000dbd <+77>: ret
0x00000000100000dbe <+78>: xchg    ax, ax
```