8.6.1

The answer can be vary, depending on different versions of the compiler

8.6.2

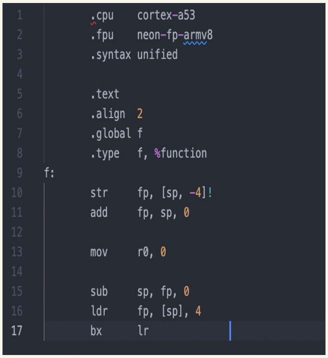
The contents of the register should be one less than the integer you entered when the program prompted you. The C statement uses 12 bytes of program memory.

9.4.1

the exit code is

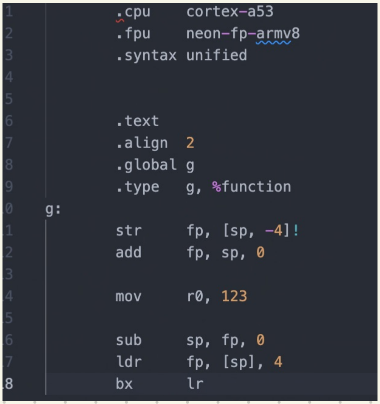
00 000 000 000 000 000 000 000 001 010 011

9.4.2



```
        .cpu    cortex-a53
        .fpu    neon-fp-armv8
        .syntax unified

        .text
        .align  2
        .global f
        .type   f, %function
f:
        str     fp, [sp, -4]!
        add     fp, sp, 0

        mov     r0, 0

        sub     sp, fp, 0
        ldr     fp, [sp], 4
        bx      lr
```

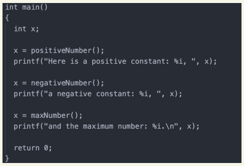
9.4.3



```
        .cpu    cortex-a53
        .fpu    neon-fp-armv8
        .syntax unified


        .text
        .align  2
        .global g
        .type   g, %function
g:
        str     fp, [sp, -4]!
        add     fp, sp, 0

        mov     r0, 123

        sub     sp, fp, 0
        ldr     fp, [sp], 4
        bx      lr
```

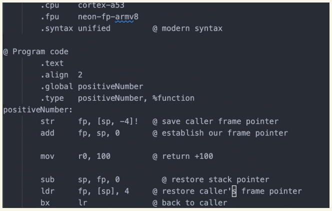
9.4.4

main



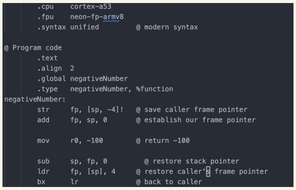
```
int main()
{
  int x;

  x = positiveNumber();
  printf("Here is a positive constant: %i, ", x);

  x = negativeNumber();
  printf("a negative constant: %i, ", x);

  x = maxNumber();
  printf("and the maximum number: %i.\n", x);

  return 0;
}
```

Posnumber



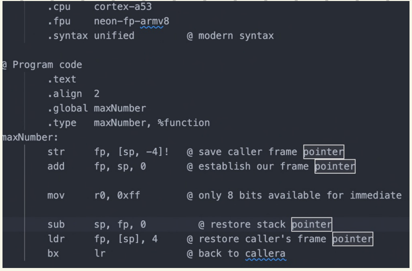
```
        .cpu    cortex-a53
        .fpu    neon-fp-armv8
        .syntax unified         @ modern syntax

@ Program code
        .text
        .align  2
        .global positiveNumber
        .type   positiveNumber, %function
positiveNumber:
        str     fp, [sp, -4]!   @ save caller frame pointer
        add     fp, sp, 0       @ establish our frame pointer

        mov     r0, 100         @ return +100

        sub     sp, fp, 0         @ restore stack pointer
        ldr     fp, [sp], 4     @ restore caller's frame pointer
        bx      lr              @ back to caller
```

Neg Number



```
        .cpu    cortex-a53
        .fpu    neon-fp-armv8
        .syntax unified         @ modern syntax

@ Program code
        .text
        .align  2
        .global negativeNumber
        .type   negativeNumber, %function
negativeNumber:
        str     fp, [sp, -4]!   @ save caller frame pointer
        add     fp, sp, 0       @ establish our frame pointer

        mov     r0, -100        @ return -100

        sub     sp, fp, 0         @ restore stack pointer
        ldr     fp, [sp], 4     @ restore caller's frame pointer
        bx      lr              @ back to caller
```

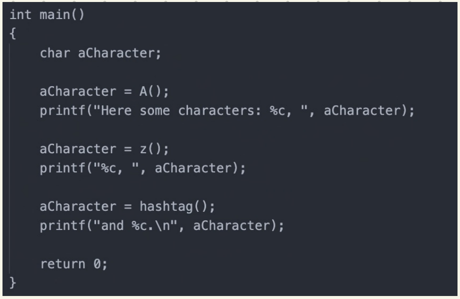
max Number



```
        .cpu    cortex-a53
        .fpu    neon-fp-armv8
        .syntax unified         @ modern syntax

@ Program code
        .text
        .align  2
        .global maxNumber
        .type   maxNumber, %function
maxNumber:
        str     fp, [sp, -4]!   @ save caller frame pointer
        add     fp, sp, 0       @ establish our frame pointer

        mov     r0, 0xff        @ only 8 bits available for immediate

        sub     sp, fp, 0         @ restore stack pointer
        ldr     fp, [sp], 4     @ restore caller's frame pointer
        bx      lr              @ back to callera
```

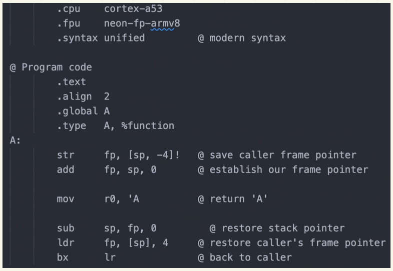
9.4.5

main



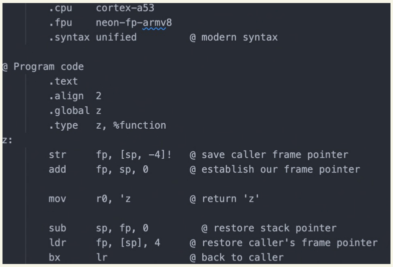
```
int main()
{
    char aCharacter;

    aCharacter = A();
    printf("Here some characters: %c, ", aCharacter);

    aCharacter = z();
    printf("%c, ", aCharacter);

    aCharacter = hashtag();
    printf("and %c.\n", aCharacter);

    return 0;
}
```

A



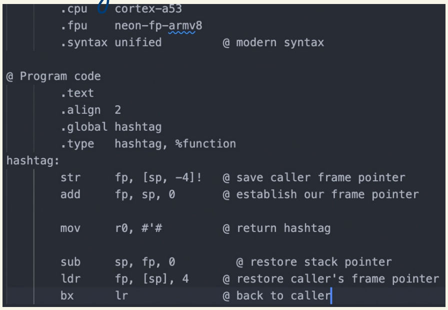
```
        .cpu    cortex-a53
        .fpu    neon-fp-armv8
        .syntax unified         @ modern syntax

@ Program code
        .text
        .align  2
        .global A
        .type   A, %function
A:
        str     fp, [sp, -4]!   @ save caller frame pointer
        add     fp, sp, 0       @ establish our frame pointer

        mov     r0, 'A          @ return 'A'

        sub     sp, fp, 0         @ restore stack pointer
        ldr     fp, [sp], 4     @ restore caller's frame pointer
        bx      lr              @ back to caller
```

Z



```
        .cpu    cortex-a53
        .fpu    neon-fp-armv8
        .syntax unified         @ modern syntax

@ Program code
        .text
        .align  2
        .global z
        .type   z, %function
z:
        str     fp, [sp, -4]!   @ save caller frame pointer
        add     fp, sp, 0       @ establish our frame pointer

        mov     r0, 'z          @ return 'z'

        sub     sp, fp, 0         @ restore stack pointer
        ldr     fp, [sp], 4     @ restore caller's frame pointer
        bx      lr              @ back to caller
```

hastag.



```
        .cpu    cortex-a53
        .fpu    neon-fp-armv8
        .syntax unified         @ modern syntax

@ Program code
        .text
        .align  2
        .global hashtag
        .type   hashtag, %function
hashtag:
        str     fp, [sp, -4]!   @ save caller frame pointer
        add     fp, sp, 0       @ establish our frame pointer

        mov     r0, #'#         @ return hashtag

        sub     sp, fp, 0         @ restore stack pointer
        ldr     fp, [sp], 4     @ restore caller's frame pointer
        bx      lr              @ back to caller
```