Section 2.7 — Assembling an ARM Program (Mazidi)

Chapter 2 · Section 2.7 — Exercises (Mazidi)

> Problems are paraphrased to respect copyright. For workflow/background, see **Mazidi, Ch. 2 §2.7**.

---

42) Assembly language is a _____ (low, high)-level language while C is a _____ (low, high)-level language.

Answer: low, **high**.
Why: Assembly maps closely to machine instructions; C abstracts the hardware.

---

43) Of C and Assembly, which is more efficient in terms of code generation (program memory used)?

Answer: Assembly (typically smaller/tighter when hand-optimized).
Why: It gives direct control over instructions. *(Modern compilers may be close, but the textbook expectation is Assembly → smaller code.)*

---

44) Which program produces the obj (object) file?

Answer: The **assembler** (for `.s/.asm` sources).
Note: For C sources the **compiler** also emits object files.

---

45) True or False. The source file has the extension "asm".

Answer: True (commonly accepted; many toolchains also use **.s/.S**).

---

46) True or False. The source code file can be a non-ASCII file.

Answer: False.
Why: Source files are **text (ASCII/UTF-8)**; non-text/binary is invalid as source.

---

47) True or False. Every source file must have an EQU directive.

Answer: False.
Why: `EQU` defines constants; it's **optional**.

---

48) Do the EQU and END directives produce opcodes?

Answer: No.
Why: They are **assembler directives (pseudo-ops)**, not CPU instructions.

---

49) Why are directives also called pseudocode/pseudo-ops?

Answer: Because they give **instructions to the assembler/linker**, not to the CPU; they **do not generate machine opcodes**.

---

50) The file with the _____ extension is downloaded into ARM Flash ROM.

Answer: **`.hex`** (Intel HEX) *(sometimes `.bin` is also used)*.

---

51) Give three file extensions produced by ARM Keil.

Answer (any three): `.obj`, `.hex`, `.lst` *(also common:* `.axf`, `.map`, `.o`*)*.

Notes for learners

- Typical build: **source (.s/.asm/.c) → object (.obj/.o) → executable (.axf) → image (.hex/.bin)**.
- **Directives** (e.g., `AREA`, `EQU`, `END`) shape assembly/placement but don't execute on the CPU.
- Keil/MDK often uses **.axf** (ELF/DWARF) for debug and **.hex** for programming the MCU.