Unit 1. Article usage: Generic reference

The way we use articles with nouns having generic reference varies according to the type of noun. More specifically article usage varies depending on whether the noun is:

countable or **non-countable**
(if countable) **singular** or **plural**

Count nouns have a natural plural form, non-count nouns do not:

device	devices
mouse	mice
information	
computing	

There is a clear contrast between non-count nouns and plural nouns, on the one hand, and count singular nouns, on the other:

Computing has changed our world.
Ø Computers have changed our world.

The computer has changed our world.

When a noun is modified (e.g. by an adjective) and the modifier is placed BEFORE the noun we still usually have a context of generic reference:

***Digital** computers have changed our world.*
***Conventional serial von-Neumann** computers have changed our world.*

However when a noun is modified by elements placed AFTER the noun, there tends to be a context of specific reference:

(The) conventional serial computers **based on the von-Neumann architecture** have changed our world.

PREmodification → generic reference → article:

Von-Neumann computers have changed our world

POSTmodification TENDS to create specific reference:

(The) computers based on the von-Neumann architecture have changed our world.

EXCEPTIONS

A few adjectives, because of their meaning, imply specific reference:

The best solutions
The first solutions
The only problems

The next generation
The last problems
The same solutions

Some non count nouns have "special" plural forms:

I like whisky (the substance).
I'd like two whiskies (two typical quantities of X).

I like wine (the substance).
I like young wines (kinds of wine).

The table below summarizes the basic rules of article usage for nouns used with generic reference:

Article usage with nouns used with generic reference
Count plural nouns → ∅ article: ∅ *Computers have changed our world* Non-count nouns → ∅ article ∅ *Computing has changed our world*
Count singular nouns require some kind of article: The computer has changed our world
Premodification does not create specific reference: Conventional serial von Neumann computers have changed our world

1. Fill in the blanks with "the", "a(n)" or "ø".

1. …… mice and …… scanners are …… input devices.

2. …… mouse is …… pointing device, but …… scanner is not.

3. …… scanner is used to input text and graphics.

4. …… scanners can be divided into …… hand-held scanners and …… desktop scanners.

5. …… hand-held scanner is quite cheap, but …… desktop scanners are not.

2. Read the through the text below and fill in the blanks with "the", "a(n)" or "ø".

High Level Languages

I. Essentially (1) ..... program is (2) ..... form of communication. Its main purpose is to communicate (3) ..... description of (4) ..... process, designed by (5) ..... programmer, to (6) ..... processor which executes it. For (7) ..... successful communication it is necessary to have (8) ..... language which (9) ..... programmers

and (10) ..... processors can understand. (11) ..... central processor of a computer can only understand (12) ..... machine code. In (13) ..... machine code each instruction is (14) ..... sequence of zeroes and ones. Although (15) ..... programmer can, with (16) ..... considerable effort, understand and write programs in machine code, it is completely unsatisfactory for the production of large and reliable programs.

II. One solution to this problem is (1) ..... translation. We can use (2) ..... language which (3) ..... programmer can understand easily and then translate (4) ..... programs written in that language into machine code for (5) ..... execution. (6) ..... natural languages, like English are not suitable for this purpose. It has been more satisfactory to design (7) ..... special languages. In fact there are many of these special languages, which are called (8) ..... *high level programming languages*.

III. 1) ..... first advantage of (2) ..... well designed high level language is that (3) ..... facilities provided by such (4) ..... language can be adapted to suit (5) ..... different application areas. For example one language can be designed for (6) ..... mathematical computing with a high numerical content, another for (7) ..... commercial applications in which large amounts of (8) ..... non-numeric information need to be processed and a third for applications in which (9) ..... computer is used to simulate another system like (10) ..... airplane.

IV. (1) .....other advantage of (2) ..... high level programming languages is that (3) ..... program is easy to read and its structure is clear. This is important because writing a large program is (4) ..... difficult intellectual task and (5) ..... programmer needs to think clearly about his\her work.

V. Finally, with (1) ..... high level languages, it is possible to include features that help (2) ..... programmer to avoid making errors and that aid (3) ..... detection of those errors which are in fact made. Since (4) ..... correctness of (5) ..... finished product is very important and (6) ..... programmer is (7) ..... human being, who has a natural tendency to make errors, the inclusion of these features is (8) ..... major advantage.

KEYS

2. Read the through the text below and fill in the blanks with "the", "a(n)" or "ø".

I. Essentially (1) **a** program is (2) **a** form of communication. Its main purpose is to communicate (3) **a** description of (4) **a** process, designed by (5) **a/the** programmer, to (6) **a/the** processor which executes it. For (7) **ø** successful communication it is necessary to have (8) **a** language which (9) **ø** programmers and (10) **ø** processors can understand. (11) **The** central processor of a computer can only understand (12) **ø** machine code. In (13) **ø** machine code each instruction is (14) **a** sequence of zeroes and ones. Although (15) **the/a** programmer can, with (16) **ø/**a considerable effort, understand and write programs in machine code, it is completely unsatisfactory for the production of large and reliable programs.

II. One solution to this problem is (1) **ø** translation. We can use (2) **a** language which (3) **a/the** programmer can understand easily and then translate (4) **ø/the** programs written in that language into machine code for (5) **ø** execution. (6) **ø** natural languages, like English, are not suitable for this purpose. It has been more satisfactory to design (7) **ø** special languages. In fact there are many of these special languages, which are called (8) **ø** high level programming languages.

III. (1) **The** first advantage of (2) **a** well designed high level language is that (3) **ø/the** facilities provided by such (4) **a** language can be adapted to suit (5) **ø** different application areas. For example one language can be designed for (6) **ø** mathematical computing with a high numerical content, another for (7) **ø/the** commercial applications in which large amounts of (8) **ø** non-numeric information need to be processed and a third for applications in which (9) **the/a** computer is used to simulate another system like (10) **an** airplane.

IV. (1) **An**other advantage of (2) **ø** high level programming languages is that (3) **the/a** program is easy to read and its structure is clear. This is important because writing a large program is (4) **a** difficult intellectual task and (5) **the/a** programmer needs to think clearly about his/her work.

V. Finally, with (1) **ø** high level languages, it is possible to include features that help (2) **the/a** programmer to avoid making errors and that aid (3) **ø/the** detection of those errors which are in fact made. Since (4) **ø/the** correctness of (5) **the** finished product is very important and (6) **the/a** programmer is (7) **a** human being, who has a natural tendency to make errors, the inclusion of these features is (8) **a** major advantage.

I.

1. "a" (a singular count noun requires some kind of article)
2. "a" (a singular count noun requires some kind of article)
3. "a"/"the" (a singular count noun requires some kind of article)
4. "a"
5. "a"/"the"
6. "a"/"the"
7. generic reference with a noun count noun → article omission
8. "a" (a singular count noun requires some kind of article)
9. generic reference with plural nouns → article omission
10. generic reference with plural nouns → article omission
11. "the" (a singular count noun requires some kind of article)
12. generic reference with a noun count noun → article omission

13. generic reference with a noun count noun → article omission
14. "a" (a singular count noun requires some kind of article)
15. "a"/"the" (a singular count noun requires some kind of article)
16. "a" or omission ("effort" may be count or non-count)

II.

1. generic reference with a noun count noun → article omission
2. "a" (a singular count noun requires some kind of article)
3. "a"/"the" (a singular count noun requires some kind of article)
4. generic reference with plural nouns → article omission/"the" (postmodification [written in that language] tends to create specific reference)
5. generic reference with a noun count noun → article omission
6. generic reference with plural nouns → article omission
7. generic reference with plural nouns → article omission
8. generic reference with plural nouns → article omission

III.

1. "the" ("first" → specific reference)
2. "a" (a singular count noun requires some kind of article)
3. "the"/omission (postmodification [provided by ...] tends to create specific reference)
4. "a" (a singular count noun requires some kind of article)
5. generic reference with plural nouns → article omission
6. generic reference with a noun count noun → article omission
7. omission/"the" (postmodification [in which ...] tends to create specific reference)
8. generic reference with a noun count noun → article omission
9. "a"/"the" (a singular count noun requires some kind of article)
10. "an"/"the" (a singular count noun requires some kind of article)

IV.

1. "An(other)" (when the word other refers to a singular count noun [advantage], it is written another)
2. generic reference with plural nouns → article omission
3. "a"/"the" (a singular count noun requires some kind of article)
4. "a" (a singular count noun requires some kind of article)
5. "a"/"the" (a singular count noun requires some kind of article)

V.

1. generic reference with plural nouns → article omission
2. "a"/"the" (a singular count noun requires some kind of article)
3. "the"/omission (postmodification [of those ...] tends to create specific reference)
4. "the"/omission (postmodification [of the ...] tends to create specific reference)
5. "the" (a singular count noun requires some kind of article)
6. "a"/"the" (a singular count noun requires some kind of article)
7. "a" (a singular count noun requires some kind of article)
8. "a" (a singular count noun requires some kind of article)