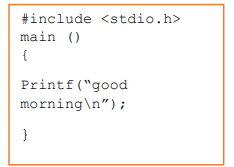
* Exercise 1: Look at the example of C program below. It tells the computer how to print the

message “good morning”. Then, discuss with your partner what you think a programming is.



-Programming is compile some code to run in the computer to execute task,solve problem or just to make command to appear on the screen

* Exercise 2: Match the words 1-5 with the definition (a-e).

1. flowchart : a diagram representing the successful logical steps of the program

2. source code : a. program instructions written in a particular computer language

3. compiler : a special program which converts the source program into machine code-the only language understood by the processor.

4.machine code : the basic instructions understood by computers, consisting of 1s and 0s (binary code).

5.debugging: the techniques of detecting and correcting errors which may occur in programs

* Exercise 3: Listen to Andrea Finch, a software developer, talking to a group of students on a training course about how a program is written. You can also check the answer of exercise 2 above.
* Exercise 4: Listen again and put these steps into the correct order

1. Understand the problem and plan a solution
2. Take a flowchart of the program
3. Write instructions in a programming language
4. Compile the program (to turn it into machine code)
5. Test and debug the program
6. Prepare documentation

- Understand the problem and plan a solution

Take a flowchart of the program

Write instructions in a programming language

Compile the program (to turn it into machine code)

Test and debug the program

Prepare documentation

* Exercise 5: Fill the missing words to complete the text. Use the words in the box

**Steps in Programming**

To write a (1) **Program** software engineers usually follow these steps. First, they try to

understand the problem and define the purpose of the program. Next, they design a step-bystep plan of instructions. This usually takes the form of a (2) **flowchart**, a diagram that uses

standardized symbols showing the logical relationship between the various parts of the

program. These logical steps are then translated into instructions written in a high-level

computer (3) **Languange** (PASCAL, COBOL, C++, etc.). These computer instructions are called the

‘source code’. The program is then (4) **Compiled** , a process that converts the source code

into machine code (binary code), the language that computers understand.

Testing program are then run to detect (5) **Errors** in the program. Errors are known as

‘bugs’, and the process of correcting these errors is called (6) **debugging** . Engineers must find the

origin of each error, then write the correct instruction, compile the program again, and

conduct another series of tests. Debugging continues until the program runs smoothly.

Finally, software developers write detailed (7) **documentation** for the users. Manuals tell us how to

use programs like word processors, databases, or web browsers

* Exercise 6: Discuss and explain each step in your own words.
* Exercise 7: Programmers sometimes use flowchart when they are planning a program. These following symbols are used in making flowchart. Identify each and its function.

|  |  |  |  |
| --- | --- | --- | --- |
| No | Symbols | Names | Functions |
| 1 |  | Terminal | Indicates the start or end of the program |
| 2 |  | Input/output | Represents input or output operations, such as reading data or displaying results. |
| 3 |  | process | Represents a process or action step, where some operation is performed. |
| 4 |  | Decision | Indicates a decision point in the program where a question is asked, and the flow branches based on the answer (e.g., yes/no). |
| 5 |  | Flowline | Shows the direction of flow from one step to the next in the process |
| 6 |  | Connector | Connects different parts of the flowchart; used to avoid crossing lines or to jump to another page. |
| 7 |  | Preparation | Represents a preparation step, such as setting a value or initializing a variable. |

* Exercise 8: Read this text carefully and then do the exercises.

Decide whether these following statements are true (T) or false (F). Then make the

necessary changes so that false statements become true.

1. A good flowchart takes into account the steps which are necessary to solve the

problem.(T)

2. It is not possible to draw a flowchart without using a template.(F)

3. There is only one possible flowchart for every problem(F)

4. Every programmer must learn flowcharting and realize its importance.(T)

5. The method of flowcharting depends on the programming language being used(F)

* Exercise 9: Flowchart David’s activities by completing the flowchart below.

David gets up in the morning, gets washed, and dressed. Before having breakfast, he checks

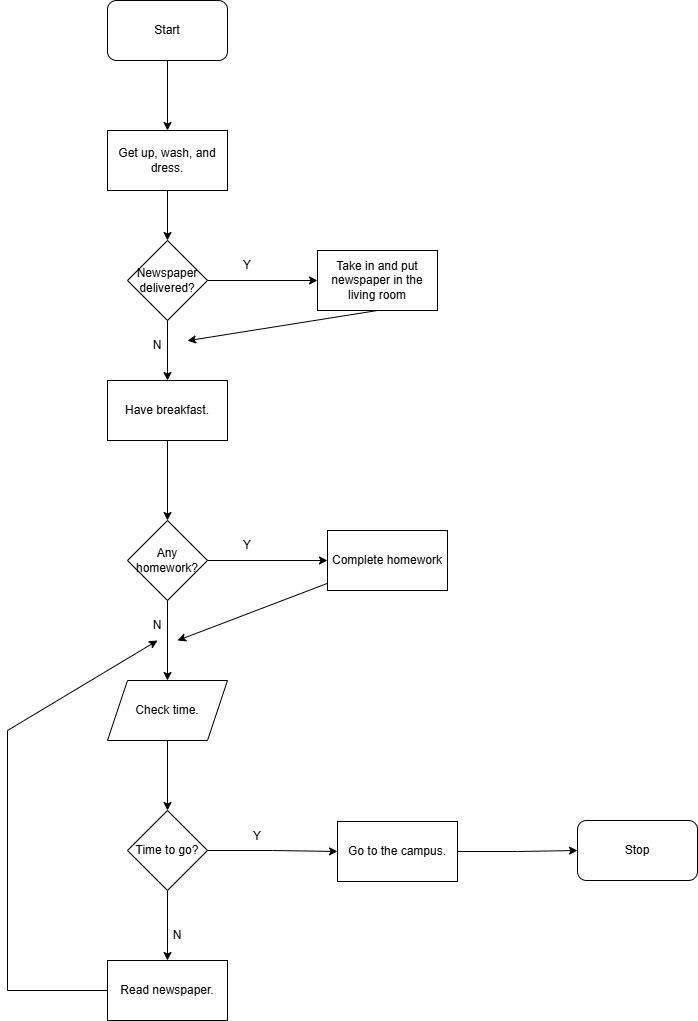
to see if the newspaper has been delivered. If it has, he takes and puts it in the living room

before sitting down to breakfast. After breakfast, he checks to make sure that he has

completed all assigned homework. If there is still some to be done, he does it. Then he checks

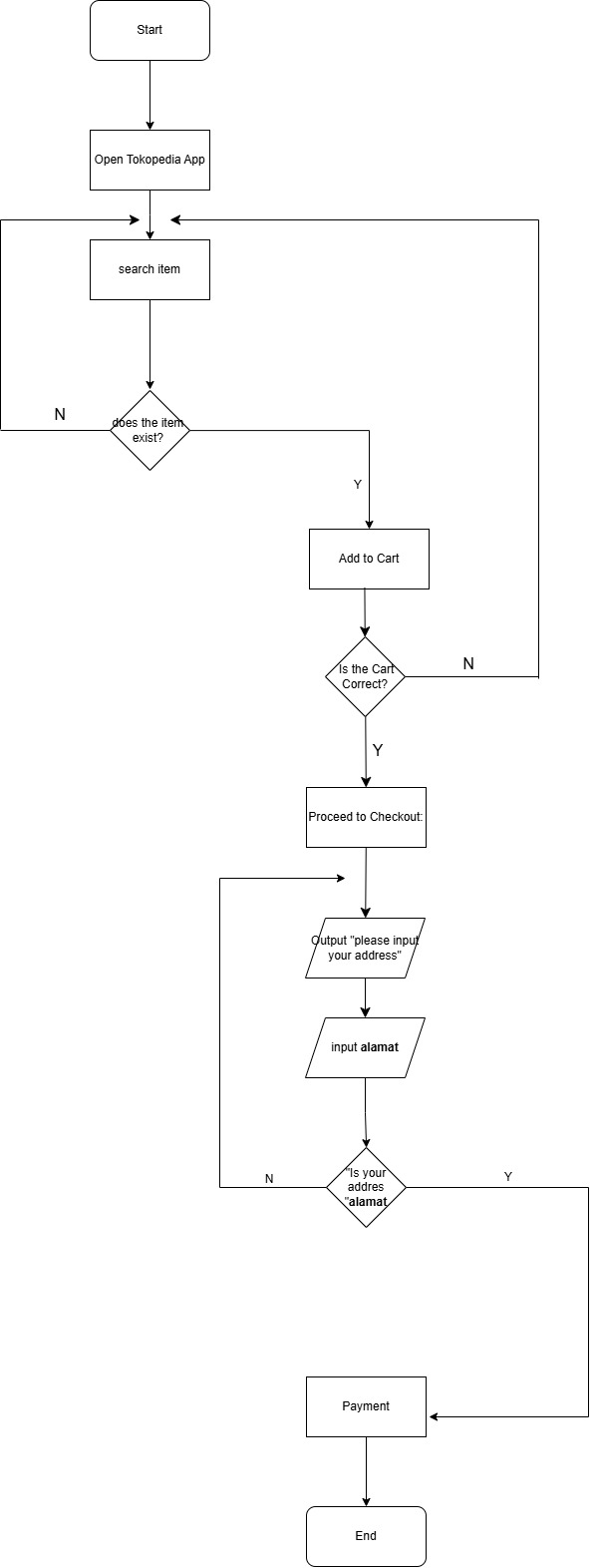
the clock, and if it is time to go, he leaves for the campus. If not, he reads the newspaper until

it is time to go



* Exercise 10: Draw a flowchart for one of these activities. Then compare your completed flowchart with other students

Buy item from Tokopedia

-

* Exercise 11: Now, write the description of the flowchart you have made on Exercise 10 and

present it in front of the class.

* Exercise 12: Find 10 words about Programming.





* Exercise 13: Read the text carefully and answer the following questions.

1. Do computers understand human languages? Why/Why not?



- No, computers do not understand human languages because they can only understand machine code, which consists of binary digits (1s and 0s). Human languages are too complex for computers to process directly.

2. What is the function of an assembler?

- The function of an assembler is to translate programs written in assembly language (a symbolic language) into machine code, which the computer can directly execute.

3. How many high-level languages are mentioned? What are they?

- FORTRAN,COBOL,BASIC,PASCAL,C (and C++),and Java

4. Why did software developers design high-level languages?

- Software developers designed high-level languages to make programming easier to write and understand, and to overcome the problem of intercommunication between different types of computers. High-level languages are closer to natural languages like English, making them more accessible to programmers.



5. What is the difference between a compiler and intepreter?

- A compiler translates the entire program's source code into machine code (object code) in one go, creating an executable file. An interpreter, on the other hand, translates the source code line by line as the program runs, without producing a separate machine code file.

6. Why are HTML and VoiceXML called markup languages?

- HTML and VoiceXML are called markup languages because they use instructions known as markup tags to format and link text files. Markup languages define how text and other content are structured and presented, rather than performing computation or logic like programming languages do.

* Exercise 14: Complete these sentences with a computer language from the text.

1. **XML** allows us to create our own tags to describe our data better. We aren’t

constrained by a pre-defined set of tags the way we are with HTML.

2. IBM developed **Fortran** in the 1950’s. It was the first high-level language in data

processing.

3. **Java** applets are small programs that run automatically on web pages and

let you watch animated characters, play games, etc.

4. **VoiceXML** is the HTML of the voice web. Instead of using a web browser and a

keyboard, you interact with a voice browser by listening to pre-recorded audio output

and sending audio input through a telephone.

5. This language is widely used in the business community. For example, the statement

ADD VAT to NET-PRICE could be used in a **Cobol** program.

* Exercise 15: Report each of these screen messages

1. Make sure the printer is switched on before continuing.

- It tells you to make sure the printer is switched on before continuing.

2. Game mode is on.

- It informs you that game mode is on.

3. Do you want to create a new document?

- It asks you if/whether you want to create a new document.

4. What is the captcha code?

- It asks you what the captcha code is.

5. Fill in your name in the box.

- It tells you to fill in your name in the box.

6. Please type the next number.

- It tells you to type the next number.

7. Enter your password.

- It tells you to enter your password.

8. Please choose from menu below.

- It asks you to choose from the menu below.

9. Can’t rename “Pictures” because a folder with that name already exists.

- It informs you that it can’t rename “Pictures” because a folder with that name already exists.

10. Exit?

- It asks you if/whether you want to exit.

11. Are you sure you want to copy the selected files?

- It asks you if/whether you are sure you want to copy the selected files.

12. Do you want to defrag the drive?

- It asks you if/whether you want to defrag the drive.

13. Mute story and posts?

- It asks you if/whether you want to mute the story and posts.

14. If you unfollow this account, you’ll have to request to follow again.

- It informs you that if you unfollow this account, you’ll have to request to follow again.

15. Click the subscribe button to follow us

- It tells you to click the subscribe button to follow them.