Information Technology Essentials — Lecture 03

Dr. Karim Lounis

Fall 2023

Recap

Last session:

- Information Technology.
- Computer.
- Computer system.
- Algorithm.
- Decidability and undecidability.
- Computability.
- Complexity.

Computers

To solve a problem using a computer, the following steps are performed:

1. W
2. W
3. W
4. W
   us
5. Fe
6. Tl
   gr
7. Tl
   ra
8. Tl

Algorithms & Complexity:

Complexity studies the difficulty of solving comptational problems by looking at their solvability, requirements (space and time), efficiency, and classification.

Space complexity looks at how much memory I need to solve the problem, whereas **time complexity** looks at how much time I need to solve the problem (focusing on wost case scenario).

Big-$\mathcal{O}$ notation: $\mathcal{O}(1)$, $\mathcal{O}(log(n))$, $\mathcal{O}(\sqrt{n})$, $\mathcal{O}(n)$, $\mathcal{O}(n \cdot log(n))$, $\mathcal{O}(n^2)$, $\mathcal{O}(n^3)$, ..., $\mathcal{O}(2^n)$, and $\mathcal{O}(n!)$.

If my algorithm takes less than a second to execute (once it is a program) and does not require space, then should I claim that my algorithm has a good complexity?

teps

put.

pro-

and

Computers

To solve a problem using a computer, the following steps are performed:

1. W
2. W
3. W
4. W
   us
5. Fe
6. Tl
   gr
7. Tl
   ra
8. Tl

Algorithms & Complexity:

Complexity studies the difficulty of solving comptational problems by looking at their solvability, requirements (space and time), efficiency, and classification.

Space complexity looks at how much memory I need to solve the problem, whereas **time complexity** looks at how much time I need to solve the problem (focusing on wost case scenario).

Big-$\mathcal{O}$ notation: $\mathcal{O}(1)$, $\mathcal{O}(log(n))$, $\mathcal{O}(\sqrt{n})$, $\mathcal{O}(n)$, $\mathcal{O}(n \cdot log(n))$, $\mathcal{O}(n^2)$, $\mathcal{O}(n^3)$, ..., $\mathcal{O}(2^n)$, and $\mathcal{O}(n!)$.

No. Time and space complexity focuse on the worst case scenario, i.e., time and space when $n \to \infty$

teps

put.

pro-

and

Computers (Exercise)

Assign the correct time complexity to the following algorithms:

- Searching for a specific number in a list of numbers.
- Computing the product of two natural numbers.
- Computing the sum of numbers in a list.
- Checking for palindromes.
- Compute the product of two matrices.
- Brute-forcing password of length n with m possible characters.
- Search for a specific element in a sorted list.

Possible time complexity: $\mathcal{O}(1)$, $\mathcal{O}(log(n))$, $\mathcal{O}(\sqrt{n})$, $\mathcal{O}(n)$, $\mathcal{O}(n \cdot log(n))$, $\mathcal{O}(n^2)$, $\mathcal{O}(n^3)$, ..., $\mathcal{O}(2^n)$, and $\mathcal{O}(n!)$..

Computers

To solve a problem using a computer, the following steps are performed:

1. We
2. We
3. We
4. We usi
5. Fee
6. Th gra
7. Th rap
8. Th

Algorithm Let us solve another problem.

Problem statement 4. We want to display the numbers from 1 to 100, but for muliples of 3, print "EN" instead of the number, for multiples of 5, print "SIA", and for the numbers that are multiples of both 3 and 5, print "ENSIA".

Let us write an algorithm that solves the problem. For consistency, let us use the following algorithm template:

Input: ...

Body: ...

Output: ...

Computers

To solve a problem using a computer, the following steps are performed:

1. We
2. We
3. We
4. We usi
5. Fee
6. Th gra
7. Th rap
8. Th

Algorithm Let us solve another problem.

Problem statement 5. Consider two variables, X and Y. Each variable is assigned a value, an integer (i.e., in $\mathcal{Z}$). Using arithmetic operations, write an algorithm that permutes the values of the two variables using a exactly 3 steps, and without using a third variable Z. For example, if X=5 and Y=9, then after running the algorithm, the result will be X=9 and Y=5.

For consistency, let us use the following algorithm template:

Input: ...

Body: ...

Output: ...

Computers

To solve a problem using a computer, the following steps are performed:

1. We analyze and understand the problem.
2. We propose a solution to the problem.
3. We write the solution as a set of finite steps, called algorithm.
4. We translate the algorithm into a program by rewriting all the steps using a programming language.
5. Feed the computer with the written program with any required intput.
6. The computer, may use another program to translate the written program into a sequence of instructions — known as machine code.
7. The computer automatically executes the produced machine code and rapidly solves the problem.
8. The computer outputs the results (output).

Computers

To solve a problem using a computer, the following steps are performed:

1. We analy
2. We propc
3. We write
4. We trans … e steps using a pr
5. Feed the … intput.
6. The comp … en program intc … e.
7. The comp … de and rapidly so
8. The computer outputs the results (output).

Program (A.k.a., Computer program)

برنامج

It is a finite sequence of instructions written in a programming language to solve a problem.

Ada Lovelace (1815-1852)

Computers

To solve a problem using a computer, the following steps are performed:

1. We
2. We
3. We
4. We usi
5. Fee
6. Th gra
7. Th rapidly solves the problem.
8. The computer outputs the results (output).

Computer program: Let us put our product calculation algorithm into a computer context:

1. Read (Input) the complete list of numbers (keybaord).
2. Use a memory location **X** to store the 1st number.
3. Take the content of **X** and multiply it with the next number on the list then save the result on **X**.
4. Repeat Step 3 until numbers there is no number left.
5. Display (output) the content of **X** (on the monitor).

Computers

To solve a problem using a computer, the following steps are performed:

1. We
2. We
3. We
4. We
   usi
5. Fee
6. Th
   gra
7. Th
   rap
8. The computer outputs the results (output).

Computer program: Let us put our product calculation algorithm into a computer context:

1. Read (Input) the complete list of numbers (keybaord).
2. Use a memory location **X** to store the 1st number.
3. Take the content of **X** and multiply it with the next number on the list then save the result on **X**.
4. Repeat Step 3 until numbers there is no number left.
5. Display (output) the content of **X** (on the monitor).

Then, we use a **Programming Language** to rewrite this steps and create our program.

Computers

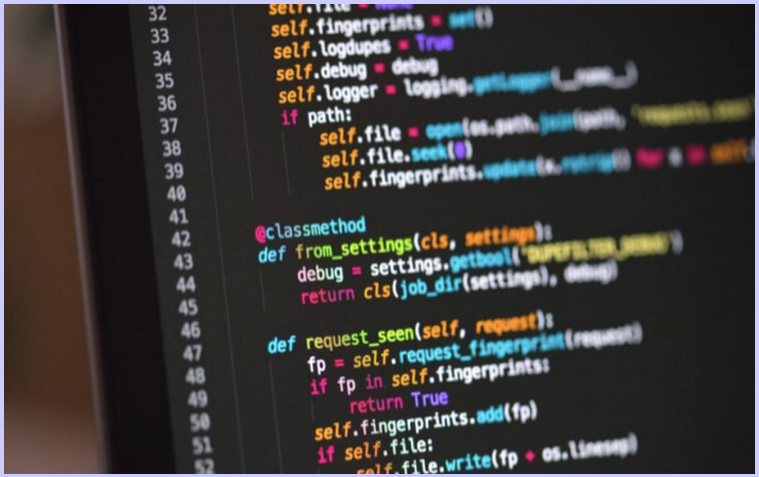
To solve a problem using a computer, the following steps are performed:

1. We analy
2. We propc
3. We write
4. We trans
   using a pr
5. Feed the
6. The comp
   gram intc
7. The comp
   rapidly so
8. The comp

Programming Language

لغة برمجة

It is a language (i.e., set of alphabets, words, grammar, and semantics) used to write computer programs.



Computers

To solve a problem using a computer, the following steps are performed:

1. We analy
2. We propc
3. We write
4. We trans
   using a pr
5. Feed the
6. The comp
   gram intc
7. The comp
   rapidly so
8. The comp

Programming Language

لغة برمجة

It is a language (i.e., set of alphabets, words, grammar, and semantics) used to write computer programs.

E.g., C, Java, C++, Python, Ruby, Ada, Pascal, etc.

Computers

To solve a problem using a computer, the following steps are performed:

1. We
2. We
3. We
4. We usi
5. Fee
6. The gra
7. The rap
8. The

Programming Language: Let us use a programming language, e.g., Pascal, to rewrite our product calculation program:

```
program ProductOfNumbers;
var
  numCount, i: integer;
  num, product: longint;
begin
  writeln('Enter the number of natural numbers:');
  readln(numCount);
  if numCount <= 0 then
  begin
    writeln('Invalid input. Please enter a positive number.');
    exit; // Exit the program
  end;
  product := 1; // Initialize the product to 1
  writeln('Enter ', numCount, ' natural numbers:');
  for i := 1 to numCount do
  begin
    readln(num);
    if num <= 0 then
    begin
      writeln('Invalid input. Please enter a positive number.');
      exit; // Exit the program
    end;
    product := product * num; // Update the product
  end;
  writeln('The product of the numbers is: ', product);
end.
```

Computers

To solve a problem using a computer, the following steps are performed:

1. We
2. We
3. We
4. We
   usi
5. Fee
6. Th
   gra
7. Th
   rap
8. Th

Programming Language: Can be classified into two categories, *declarative* and *imperative* (& some support both styles of use):

- **Declarative.** You express the desired result or outcome rather than specifying the step-by-step process to achieve that result. E.g., SQL, HTML, CSS, Prolog, Haskell, etc.
- **Imperative** You explicitly list the sequence of commands or steps to perform and achieve a specific task. E.g., C, C++, C♯, Java, Python, Ruby, Perl, Fortran, Assembly, etc.

Programming language are also classified into **high-level** and **low-level** programming language. The more we get closer to the hardware the lower is the level of the programming language.

Low-level programming languages are hardware (architecture) dependant. E.g., Assembly and binray. High-level PLs are not.

Computers

To solve a problem using a computer, the following steps are performed:

1. We
2. We
3. We
4. We
   usi
5. Fee
6. Th
   gra
7. Th
   rap
8. Th

Programming Language: Can be classified into three other categories, *complied*, *interpreted*, and *hybrid*:

- **Compiled.** The entire source code is compiled into a binary for execution (compiled once). E.g., C, C++, D, OCaml, Go, Haskell, Pascal, Fortran, Ada, COBOL, Rust, etc.
- **Interpreted** In the source code, instructions are interpreted and then executed line by line (by an interpreter) . E.g., JavaScript, HTML, Ruby, PHP, Bash, MATLAB, python, etc.
- **Hybrid.** The entire source code is compiled into an intermediary code (a.k.a., bytecode), and then interpreted further for execution. E.g., C♯, Java, Scala, objective-C, Ruby, Python, Kotlin, Groovy, Swift, etc.

Computers

To solve a problem using a computer, the following steps are performed:

1. We
2. We
3. We
4. We usi
5. Fee
6. Th gra
7. Th rap
8. Th

Machine code (A.k.a., binary code)

الشفرة الثنائية

It is the language that the computer understands.

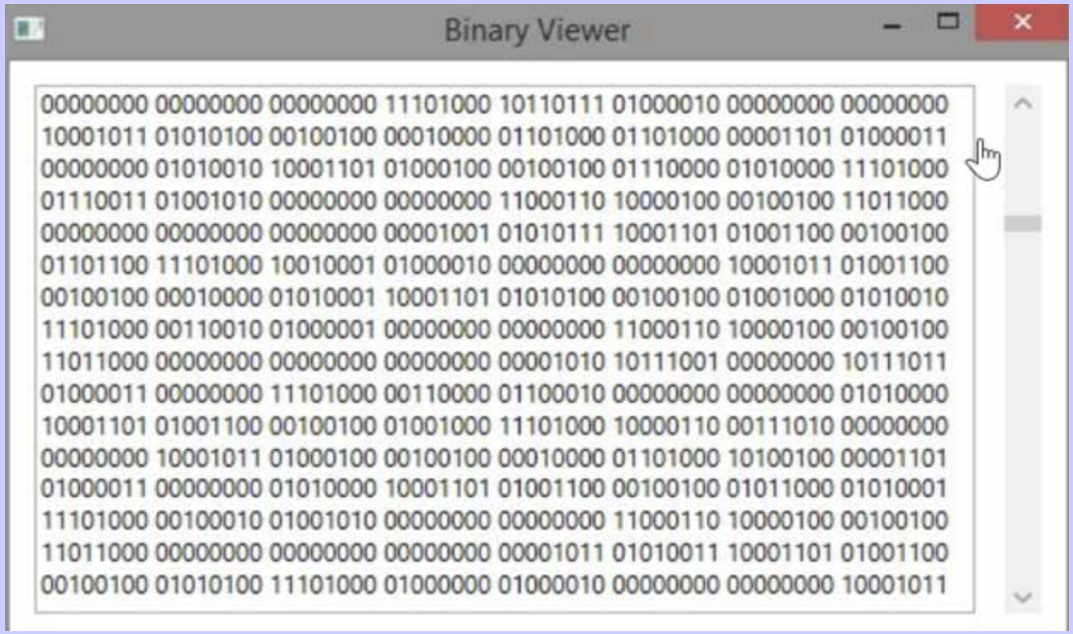
Everything is encoded in binary (0 | 1) and every sequence of bits has a semantics.

Computers

To solve a problem using a computer, the following steps are performed:

1. We
2. We
3. We
4. We usi
5. Fee
6. The gra
7. The rap
8. The computer outputs the results (output).

Machine code (A.k.a., binary code): Our product computation code would look like this:



Computers

To solve a problem using a computer, the following steps are performed:

Binary System: Computers operate on the binary system (i.e., Base 2):

- The unit element is called, a bit, which value is 0 or 1.
- The bit 0 is given voltage v, where $v \in [0\ 30\%\underline{vdd}]$ and the bit 1 is given a voltage $v \in [75\%\underline{vdd}\ \underline{vdd}]$ ($\underline{vdd}$ being the suppiled voltage).
- A sequence of 8 bits is called a Byte (or Octet).
- When the number of bits is 2^{10}=1024, we say 1Kb (Kilo).
- When the number of bits is 2^{20}, we say 1Mb (Mega).
- When the number of bits is 2^{30}, we say 1Gb (Giga).
- When the number of bits is 2^{40}, we say 1Tb (Tera).
- We can use power of 2 (i.e., 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, etc) to express all natural numbers.

Computers

To solve a problem using a computer, the following steps are performed:

Binary System: Computers operate on the binary system (i.e., Base 2):

- Computers use the binary system because of its simplicity and compatibility with electronic components.
- It simplifies the design of electronic components.
- Binary signals are less susceptible to noise and errors.
- Many electronic components, e.g., transistors and logic gates, can operate reliably with binary singals.
- Binary data can be stored using electronic components that can maintain two distinct states.
- We use the hexadecimal system (base 16) to express large binary numbers in a compact form: 10 is A (0xa), ..., and 16 is F (0xf).

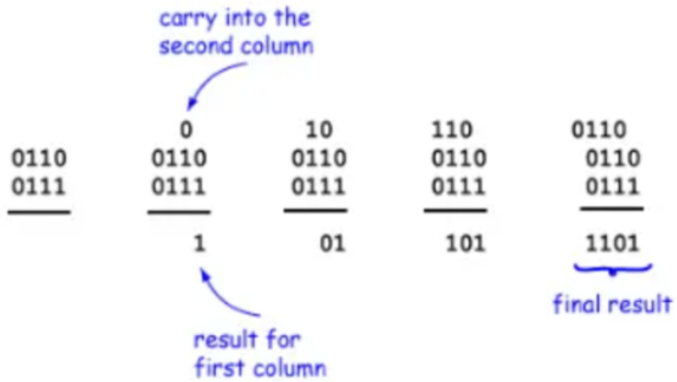
Computers

To solve a problem using a computer, the following steps are performed:

Binary System: Computers operate on the binary system (i.e., Base 2):

What is Binary Addition?

Binary Addition Example




```
    1  1  1
+   1  1  0  1
    1  1  1  1
-------------
 1  1  1  0  0
```

A + B	SUM	CARRY
0 + 0	0	0
0 + 1	1	0
1 + 0	1	0
1 + 1	0	1

Computers

To solve a problem using a computer, the following steps are performed:

Binary System: Computers operate on the binary system (i.e., Base 2):

What is Binary Subtraction?

- Two binary numbers are subtracted by subtracting each pair of bits together with borrowing, where needed.
- Subtraction Example:

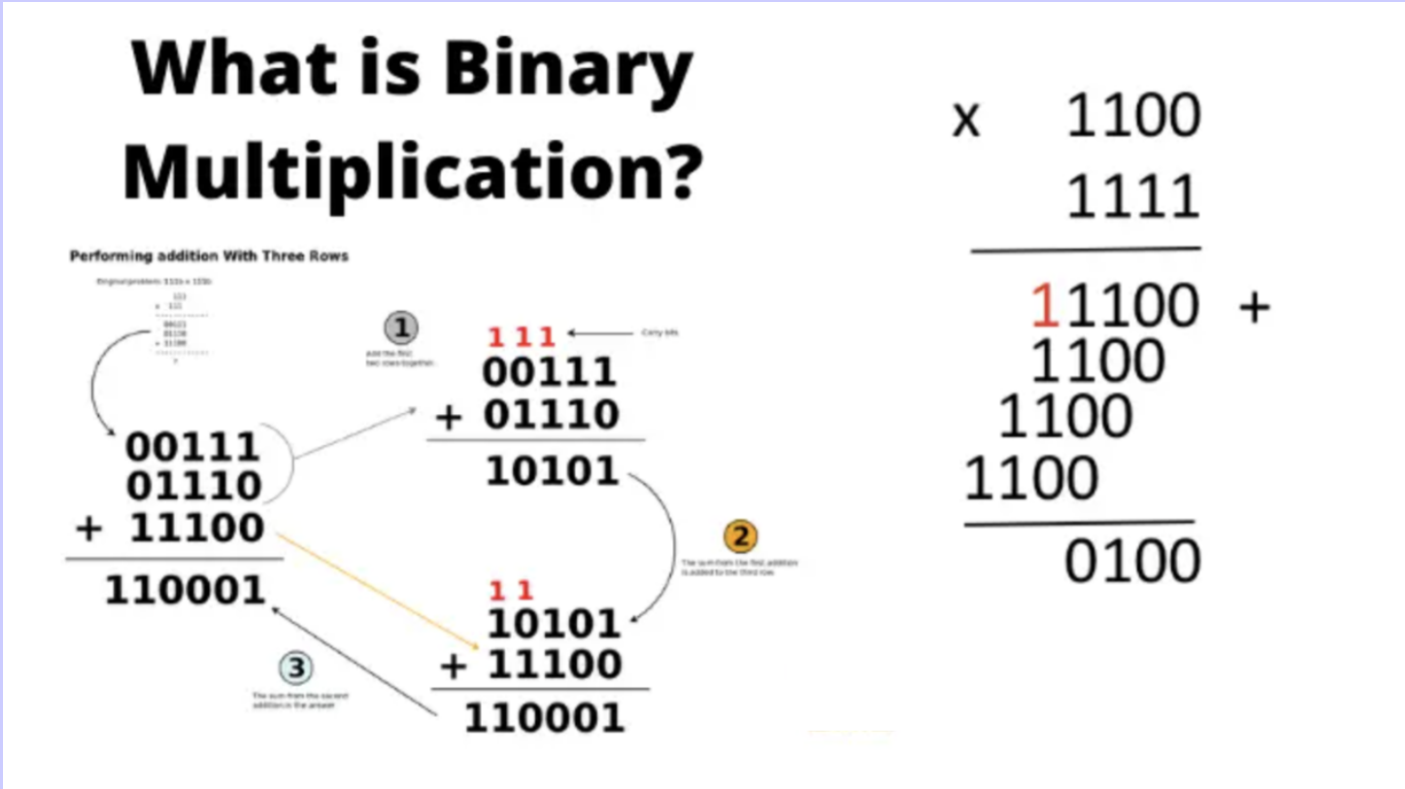
			0	0	1	1	1	1	0	0	Borrow
X	229		1	1	1	0	0	1	0	1	
Y -	46	-	0	0	1	0	1	1	1	0	
	183		1	0	1	1	0	1	1	1	

$$\begin{array}{r} 0\,1\,2 \\ 1\,1\,0\,0\,1_2 \\ -\;1\,0\,1\,1\,1_2 \\ \hline 0\,0\,0\,1\,0_2 \end{array}$$

Computers

To solve a problem using a computer, the following steps are performed:

Binary System: Computers operate on the binary system (i.e., Base 2):



Computers

To solve a problem using a computer, the following steps are performed:

Binary System: The system is used to implement the ASCII (American Standard Code for Information Interchange — 1960) character encoding standard. It encodes characters on 7 bits. E.g., 'A' is 65.

ASCII was extended to 8 bits to allow more possible encoding (1981).

While ASCII was suitable for representing English text, it lacked support for many non-English languages and characters.

Used (and still being used) for encoding characters of text files.

It is backward compatible and used in various networking protocols (e.g., HTTP) and programming languages. It is also used in character devices.

How would a computer encode the term ensia in ASCII-128?

Computers

To solve a problem using a computer, the following steps are performed:

Binary System: The system is used to implement the ASCII (American Standard Code for Information Interchange — 1960) character encoding standard. It encodes characters on 7 bits. E.g., 'A' is 65.

ASCII was extended to 8 bits to allow more possible encoding (1981).

While ASCII was suitable for representing English text, it lacked support for many non-English languages and characters.

Used (and still being used) for encoding characters of text files.

It is backward compatible and used in various networking protocols (e.g., HTTP) and programming languages. It is also used in character devices.

How would a computer encode the term ensia in ASCII-128?

101_{10} 110_{10} 115_{10} 105_{10} 97_{10}

Computers

To solve a problem using a computer, the following steps are performed:

Binary System: The system is used to implement the ASCII (American Standard Code for Information Interchange — 1960) character encoding standard. It encodes characters on 7 bits. E.g., 'A' is 65.

ASCII was extended to 8 bits to allow more possible encoding (1981).

While ASCII was suitable for representing English text, it lacked support for many non-English languages and characters.

Used (and still being used) for encoding characters of text files.

It is backward compatible and used in various networking protocols (e.g., HTTP) and programming languages. It is also used in character devices.

How would a computer encode the term ensia in ASCII-128?

$01100101_2\ 01101110_2\ 01110011_2\ 01101001_2\ 01100001_2$

Computers

To solve a problem using a computer, the following steps are performed:

Bin
Sta
star

ASC

Wh
for

Use

It is
HT

Hov

merican
coding

981).

upport

s (e.g.,
levices.

8?

What about other languages (e.g., Semitic languages), math symbols, Greek symbols, emojies, etc?

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Computers

To solve a problem using a computer, the following steps are performed:

Binary System: The system is used to implement the UNICODE (UNIversal CODE — 1990s) character encoding standard. It assigns unique codes [U+0000 to U+10FFFF] to each character. I.e., from 0 to 1,114,111. E.g., 'A' is U+0041, which corresponds to 65 in decimal.

UNICODE aims to encode all the world's written scripts and characters (e.g., symbols, emojis, mathematical symbols, etc).

It defines 3 different encoding schemes: UTF-8, UTF-16, and UTF-32, to encode all UNICODE code points (Unicode Transformation Format).

The most commonly used is UTF-8.

It has played a crucial role in enabling the globalization and internationalization of digital content and applications.

Computers

To solve a problem using a computer, the following steps are performed:

1. We analyze and understand the problem.
2. We propose a solution to the problem.
3. We write the solution as a set of finite steps, called algorithm.
4. We translate the algorithm into a program by rewriting all the steps using a programming language (producing the source code).
5. Feed the computer with the written program with any required intput.
6. The computer, may use another program to translate the written program into a sequence of instructions — known as machine code.
7. The computer automatically executes the produced machine code and rapidly solves the problem.
8. The computer outputs the results (output).

Computers

To solve a problem using a computer, the following steps are performed:

1. We analyze and understand the problem.
2. We propose a solution to the problem.
3. We write the solution as a set of finite steps, called algorithm.
4. We translate the algorithm into a program by rewriting all the steps using
5. Feed the computer with the written program with any required intput.
6. The computer, may use another program to translate the written program into a sequence of instructions — known as machine code.
7. The computer automatically executes the produced machine code and rapidly solves the problem.
8. The computer outputs the results (output).

How do we call the program used in Step 6?

Computers

Definition

Compiler. It is a program used to translate a computer program written in a programming language into another computer program written in a lower-level (or same-level) programming language.

Among various tasks, the compiler does the following:

- **Syntax Analysis.** Checks for syntax errors — perform parsing.
- **Semantics Analysis.** Checks the semantics of the code.
- **Optimization.** This would improve the efficiency of the code.
- **Code Generation.** Generate the lower-level code.

Actually, the overall process of compilation goes through three main stages: (1) Compilation, (2) Assembly, and (3) Linking.

Computers

Definition

Compil
in a prc
lower-lev

Among

- **Syn**
- **Sem**
- **Op**
- **Co**

Compilation: The overall process goes through:

1. **Compilation.** Translate high-level source code (e.g., *.c or *.cpp) into low-level code, generally, assembly (i.e., *.asm).
2. **Assembly.** Translate the generated assembly code into machine code (object files — *.obj). It also translate all referenced library files into object files.
3. **Linking.** Combine multiple object files and libraries to create the executable file, i.e., *.exe or *.out.

Actually, the overall process of compilation goes through three main stages: (1) Compilation, (2) Assembly, and (3) Linking.

Computers

How do I write my proper program?

The following steps would generally make it:

1. Understand the problem for which you want to write a program.
2. Design and write an algorithm that solve the problem.
3. Choose a programming language and a text editor.
4. Writre your code by translating the algorithm using the choosen programming language's syntax and grammar.
5. Compile the program you wrote using a compiler to generate the binary file, while making sure that all used libraries are available in ur pc.
6. Execute (or Debug) ur program for some input & retrieve the results.

There exist some tools that gather the text editor, the compiler, and the executer, all in one software application, called **IDE (Integrated Development Environment)**. E.g., Eclipse, Visual Studio, PyCharm, NetBeans, Visual Studio Code, and Code::Blocks.

Computers

How do I write my proper program?

The follo[illegible] steps would generally make it

1. Un[illegible]
2. Des[illegible]
3. Ch[illegible]
4. Wr[illegible] gra[illegible]
5. Co[illegible] file[illegible]
6. Exe[illegible]

There e[illegible] executer[illegible] **opment** Visual Studio Code, and Code::Blocks.

Text editor: It is an application software that is used to write plain text files. E.g., Notepad++, gedit, nano, vim, bluefish, etc.

You could also use some online web applications that allow you to write code, compile, and execute. E.g.,

https://www.onlinegdb.com/
https://www.tutorialspoint.com/codingground.htm
https://onecompiler.com/

Another option, not recommanded for novice coders, is to use generative AI systems such as **ChatGPT**. Such systems can generate the code you want in a matter of few seconds.

Computers

How do I write my proper program?

The follo

1. Un
2. Des
3. Ch
4. Wr gra
5. Co file
6. Exe

There e executer **opment** Visual S

Let's write our 1st "**You are a great student**" using HTML:

1. Algorithmically, we want to display a message.
2. Open a text editor, e.g., notepad.
3. Write the following lines of code:

```
1 <html>
2 <head>
3 <title>My first interpreted program</title>
4 </head>
5     <body>
6         <h1>You are a great student</h1>
7     </body>
8 </html>
```

4. Save your file with extension **.html** or **.htm**.
5. Use your Internet Browser (Gooogle Chrome, IE, Safari, Firefox, etc) to open your file.

Computers (Exercise)

Read the following definitions and find the right term that fits with each definition:

- A numbering system used to generate machine code.
- A language that computers understand and use to execute programs.
- A language that is used to write computer programs.
- A set of instructions written in a programming language.
- A program that translate user's computer programs into machine code.
- A set of instruction written in natural language to solve a problem.
- Cannot be solved (computed) by a computer in a finite amount of time.

Program, algorithm, undecidable problem, machine code, programming language, compiler, and binary.

- End.