1. **2016-1**

   Circle or cross: "**T**" if True – "**F**" if False.

   T / F   A service-mark is a mark to identify a service rather than a product. For example MGM (Metro-Goldwyn-Mayer) uses the sound of a lion's roar.

   T / F   Regular Expression:

   ```
   ^.[aiueo].[aiueo]$
   ```

   will match this following string:

   ```
   sate satu
   ```

   T / F   Free software is always free of charge.

   T / F   Open Source Software is not always free of charge.

   T / F   There are many Free Software licenses, however, there is only one copyleft license, i.e. the GNU / General Public Licenses.

   T / F   The output of script:

   ```
   echo \a b c d"  |  echo \a b c d"
   ```

   is

   ```
   a b c d
   a b c d
   ```

   T / F   The output of script:

   ```
   for II in A B C
   do
         echo X$II
   done
   ```

   is

   ```
   A
   B
   C
   ```

2. **2016-2a**

   Circle or cross: "T" if True – "F" if False.

   T / F   EULA (End User License Agreement) is an example of a non-free software license.

   T / F   You don't own the propriety software you have bought.

   T / F   Public Domain software is not Free Software.

   T / F   Free Software is not always Copy-left.

   T / F   "[ˆc]at" matches all strings matched by "‗at" except "cat".

   T / F   "[a-z0-9]" matches any single letter or any single digit.

3. **2016-2b**

   These following are some scripting examples. Fill remaining empty "output" cells.

Script (date=19 Oct 2016)	Output
`echo "1 2 3 4 5"	awk '{print $1 " " $5}'`	
`date +"%d %b %Y"`	
`echo \abc"	tr '[a-z]' '[A-Z]'`	
`echo 0123456789	cut -c5-9`	

`VAR="hallo"`
`case "$VAR" in`
`  hallo) echo "Hallo too!"`
`       ;;`
`  *)    echo "What?"`
`       ;;`
`esac`
`exit 0`	

`date +"%d %b %Y"	awk '{print $2 " " $1 ", " $3}'`	
`% ID="VWXYZ"` `% echo "$ID"	cut -c1-3	tr '[a-z]' '[A-Z]'`	

`VAR="hello"`
`case "$VAR" in`
`  hallo) echo "Hallo too!"`
`       ;;`
`  *)    echo "What?"`
`       ;;`
`esac`
`exit 0`	

4. **2017-1**

Circle or cross: "T" if True – "F" if False.

T / F   Free Software means "software you can get for free (gratis)."

T / F   According to the Free Software Movement, free software developers should never be paid.

T / F   Free Software never has a license.

T / F   The Open Source Initiative (OSI) agrees with the Free Software Foundation (FSF) about how
to promote the (Free or Open Source) software.

T / F   Both Free Software Licenses and Open Source Licenses may not discriminate against anyone.
Giving everyone freedom means giving evil people freedom, too.

T / F   You don't own software, you just get a license to use software.

T / F   Public Domain Software is Free Software.

T / F   The Free Software Movement hates Microsoft and consider it the Great Satan.

T / F   Copyleft is method to keep a free-software – and its modification – to be free.

T / F   Most free software projects are developed by a single developer (or with no contributor).

Shell Script

```
001 #!/bin/bash                                          023
002 # (c) 2017-2019 Rahmat M. Samik-Ibrahim              024 ONE='1'
003 # This is a free script, lah!                        025 TEN='10'
004 # REV02: Tue Feb 19 19:27:58 WIB 2019                026 REGEX='^[0-9]+$'
005 # START: Wed Mar 29 17:28:20 WIB 2017                027 VARIABLE='1234567892'
006 # ==============================================      028 HASH=`echo "$VARIABLE" | sha1sum`
007 # cut     remove sections from each line of files     029 echo        "$HASH"
008 # cut -c  select only these characters                030 isADigit
009 # sha1sum compute and check SHA1 message digest       031 TEXT2=`echo "$HASH"  | cut -c1-3`
010 # tr      translate or delete characters              032 TEXT3=`echo "$TEXT2" | tr '[a-z]' '[A-Z]'`
012 isADigit() {                                          033 echo "$TEXT1"
013    [[ $VARIABLE =~ $REGEX ]] || {                     034 echo "$TEXT2"
014       TEXT1="VARIABLE $VARIABLE is not a digit."       035 echo "$TEXT3"
015       return                                          036 exit 0
016    }
017    [[ ${#VARIABLE} =~ $TEN ]] && {
018       TEXT1="VARIABLE $VARIABLE has 10 digits."
019       return
020    }
021    TEXT1="VARIABLE $VARIABLE has no 10 digits."
022 }
```

Script Output (Line 029, 033, 034, 035):

```
5a3ce516f19aa888e8fd3e80e7d7cb0320253d25  -
```

5. **2017-2**

According to the Free Software Foundation (FSF), (01) terms allow the software users to run the software for any purpose as well as to study, change, and distribute the software and any adapted versions. (02) software has absolutely no ownership such as copyright, trademark, or patent. (03) software requires that information necessary for reproducing and modifying the work must be made available to recipients of the software. There is a difference between the terms (04) and (05): it is possible to use (06) code in (07) projects, but the inverse is not allowed. (08) protect the expression of ideas, whereas, (09) is a right to exclude others from making, using, or selling an invention.

Match the number of the sentence above with these following phrases:

[   ] Copyleft   [   ] Copyrights   [   ] Free Software   [   ] Free Software

[   ] Free Software   [   ] Open Source   [   ] Open Source   [   ] Public Domain   [   ] Patent

What will be the output of these following "sed" commands? Consider, the content of `thefile.txt` is:

```
Sukma's house is 10 km away.
```

```
$ sed -e s/"km "/"miles "/ thefile.txt
```

```
$ sed -e s/"km "/"miles "/g thefile.txt
```

```
$ sed -e s/km/miles/g thefile.txt
```

```
$ sed -e s/[km]/miles/g thefile.txt
```

----

6. **2018-1**

(01) is the most famous open-source operating system, while (02) is a well-known example of the opposite closed-source approach. The Free Software Foundation (03) the software to encourage sharing rather than (04) it. An Operating System provides certain services including user-interfaces like: (05) interface, (06) interface, and (07) interface. A (08) is a program designed to be run by a command-line interpreter. The (09) or (10) is an example of a command-line interpreter.

Match the number of the sentence above with these following phrases:

[   ] bash   [   ] copyright   [   ] Bourne-Again shell   [   ] Command-Line   [   ] copylefts

[   ] Batch   [   ] Graphical User   [   ] Microsoft Windows   [   ] shell script   [   ] Linux

What is the output of this following shell-script:

```
#!/bin/bash
# (c) 2018 This's a free script
echo "START"
for II in "SATU DUA TIGA"
do echo $II ; done
for II in SATU DUA TIGA
do echo $II ; done
echo "STOP"
```

7. **2018-2 (81%)**

Both (01) operating systems and (02) operating systems are available in source-code format. Free software not only makes (03) available but also is licensed to allow (04), (05) and (06). (07), (08) and (09) are examples of popular open-source systems. An advantage of working with open-source operating systems is their (10).

Match the number of the sentence above with these following phrases:

[   ] diversity (50%)   [   ] free (100%)   [   ] FreeBSD (100%)   [   ] GNU/Linux (100%)

[   ] modification (100%)   [   ] no-cost use (50%)   [   ] open-source (100%)   [   ] redistribution (90%)

[   ] Solaris (100%)   [   ] source code (80%)   [   ] —   [   ] —

What is the output of this following shell-script (74%):

```
#!/bin/bash
# (c) 2018 This's a free script
VAR="talas"
echo "$VAR"
KALIMAT="Keladi tidak sama dengan $VAR"
echo "$KALIMAT"
echo "$KALIMAT" | gawk '$5 ~ /s$/ {print $1}'
```

8. **2019-1 (74.3%)** (Ref: Schilberschatz et.al.)

    (01) is a software that is licensed on a subscription basis.

    (02) is a software stack ready for application use.

    (03) provides hardware, storage, servers, and software over the network.

    (04) response should be guaranteed within a specified timing constraint.

    (05) is an interface for user-level processes to request services from the kernel of an operating system.

    (06) is also called a Virtual Machine Manager (VMM).

    (07) allows Operating Systems to run as applications within other Operating Systems.

    (08) allows applications compiled for one CPU to run on another CPU.

    (09) is a type of computing that delivers infrastructures, platforms, and applications "as a service" across a network.

    "Free Software" refers to freedom of use, not (10). Both (11) operating systems and (12) operating systems are available in source-code format rather than as compiled binary code.

    Match the number(s) in the sentence above with these following phrases:

    [ ] CLOUD COMPUTING (77%) [ ] EMULATION (66%) [ ] FREE (87%) [ ] HYPERVISOR (77%)

    [ ] IAAS (66%) [ ] OPEN-SOURCE (88%) [ ] PAAS (66%) [ ] PRICE(89%)

    [ ] REAL TIME SYSTEM (87%) [ ] SAAS (62%) [ ] SYSTEM CALLS (80%) [ ] VIRTUALIZATION (64%)

    What is the output of this following shell-script (75%):

    ```
    #!/bin/bash
    # (c) 2019 0324 2110 This is a free script
    TIGA=3
    ANGKA="1 2 \$TIGA"
    echo "START"
    echo "ANGKA"
    echo "$ANGKA"
    echo "$ANGKA" | gawk '/TIGA/ {print $1, $2}'
    echo "STOP"
    ```

9. **2019-2 (57.1%)**

    (01) is a software that is licensed on a subscription basis.

    (02) is a software stack ready for application use.

    (03) provides hardware, storage, servers, and software over the network.

    (04) response should be guaranteed within a specified timing constraint.

    (05) is an interface for user-level processes to request services from the kernel of an operating system.

    (06) is also called a Virtual Machine Manager (VMM).

    (07) allows Operating Systems to run as applications within other Operating Systems.

    (08) allows applications compiled for one CPU to run on another CPU.

    (09) is a type of computing that delivers infrastructures, platforms, and applications "as a service" across a network.

    Free software is sometimes referred to as (10) software. MacOS contains a(n) (11) kernel named Darwin. Microsoft Windows is (12) software.

Match the number(s) in the sentence above with these following phrases:

[   ] CLOUD COMPUTING (76%)   [   ] EMULATION (17%)   [   ] OPEN-SOURCE (20%)   [   ] HYPERVISOR (80%)

[   ] IAAS (71%)   [   ] PROPRIETARY (47%)   [   ] PAAS (66%)   [   ] LIBRE (27%)

[   ] REAL TIME SYSTEM (83%)   [   ] SAAS (56%)   [   ] SYSTEM CALLS (77%)   [   ] VIRTUALIZATION (13%)

What is the output of this following shell-script (62%):

```
001 #!/bin/bash
002 # (c) 2019 10251314 This is a free script
003
004 LINE1="= BEGIN"
005 LINE2="TESTING regex using awk"
006 LINE3="TESTING REGEX using awk"
007 LINE4="= = END"
008
009 echo $LINE1
010 echo $LINE2 | awk '/regex/{print $1}'
011 echo $LINE3 | awk '/regex/{print $1}'
012 echo $LINE2 | tr [A-Z] [a-z] | awk '/regex/{print $1}'
013 echo $LINE3 | tr [A-Z] [a-z] | awk '/regex/{print $1}'
014 echo $LINE4
015
016 exit
```

_____

_____

_____

_____

_____

_____

_____

_____

10. **2020-1/2022-2 (43%)**

```
001 #!/bin/bash
002 # (c) 2020 0310Tue1434  This is a free script.
003 # FileName: 01-Week01.sh
004 function INFO() {
005    if [ -z "$1" ]
006    then
007       printf "Script Name: $SCRIPT\n"
008    else if [ -e "$1" ]
009       then
010          RESULT=$(grep "$PREFIX" "$1")
011          if [ ! -z "$RESULT" ]
012          then
013             printf "[$SCRIPT] $RESULT\n"
014          fi
015       fi
016    fi
017 }
```

```
018 PREFIX="^# FileName: "
019 SCRIPT="$0"
020 INFO "$1"
021 exit 0
```

Consider script file "01-Week01.sh".
What is the output of:

(a) (58%)

    $ bash 01-Week01.sh

_____

(b) (38%)

    $ bash 01-Week01.sh 01-Week01.sh

_____

11. **2023-2 (AW:98%)**

Fill in the description number that matches the description with its corresponding regular expression.

No.	Description
1.	Validate that the input consists of alphanumeric characters and underscores, with a minimum and maximum length (100%)
2.	Validate that the input is a valid date in the format "YYYY-MM-DD." (100%)
3.	Numeric Input Validation Ensure that the input consists only of numeric characters. (100%)
4.	Validate that the input is a valid email address. (100%)
5.	Validate that the input is a valid file name with a specific file extension (e.g., .jpg, .png). (100%)
6.	Validate that the input is a valid URL path with no blank characters (e.g., "/products/123" or "/user/profile"). (100%)
7.	Validate that the input is a valid time in the format "HH:MM" (24-hour clock). (100%)
8.	Validate that the input does not contain HTML tags. (100%)
9.	Validate that the input represents a valid age (e.g., between 1 and 120). (100%)
10.	Validate that the input is a valid full name, allowing alphabetic characters, spaces, and common special characters like hyphens and apostrophes. (100%)

Description No	Regular Expression
	`/^[0-9]+$/`
	`/^[A-Za-z\s'-]+$/`
	`/^\d{4}-(0[1-9]|1[0-2])-(0[1-9]|[12]\d|3[01])$/`
	`/^(?:[1-9][0-9]?|1[01][0-9]|120)$/`
	`/^\/[a-zA-Z0-9\-_\/]*$/`
	`/^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/`
	`/^(.+\.(jpg|jpeg|png|gif))$/`
	`/^[a-zA-Z0-9_]{3,20}$/`
	`/^(0[0-9]|1[0-9]|2[0-3]):[0-5][0-9]$/`
	`/<[^>]*>/`