Introduction to Computer Science: Final Exam

January 10, 2020. 9:30 - 11:00

Name: _____

Student ID: _____

Instructions: This paper consists of 50 multiple choice questions. You have to answer all of them. Each carries two marks. Each question has only one correct option. Please put down the answer clearly next to the question number.

The information regarding the artificial CPU and the source codes of the programs for answering the questions are put in the end of the paper.

Reminder: Beware of careless mistake!

Question 1

If the following C program segment is executed, what will you see on the screen?

```
i=0;
While (i<3)
{ printf("***\n");
  i = i+1;
}
printf("%d", i);
```

(a) *********3
    c:\>

(b) *********
    3
    c:\>

(c) ***
    ***
    ***3
    c:\>

(d) ***
    ***
    ***
    3
    c:\>

(e) None of the above.

Question 2

What will you see on the console if the following program segment is executed?

```
#include<stdio.h>
#include<stdlib.h>

main()
{
int i;

for(i=4; i>0; i--)
printf("%d", i%4);
printf("%d", i);
}
```

(a) 0 1 2 3 4
    c:\>

(b) 0 1 2 3 4

    c:\>

(c) 4 3 2 1 0
    c:\>

(d) 4 3 2 1 0

    c:\>

(e) None of the above.

Question 3

Refer to Program A. What is the content of "argv[0]" if the following command is typed?

```
c:\>winnerloser w 2 3 8 4 7 9 11 2 1 33
```

(a) winnerloser

(b) w

(c) 2

(d) 3

(e) 12.

Question 4

Refer to Program A. What is "atoi()" used for?

(a) To convert the string argv[2] to an integer number.

(b) To convert the string argv[2] to an array of integer numbers.

(c) To convert the string argv[2] to a floating point number.

(d) To convert the string argv[2] to an array of floating point numbers.

(e) None of the above.

Question 5

Refer to Program A. What will you see if the following command is typed?

```
c:\>winnerloser w 2 3 8 4 7 9 11 2 1 33
```

(a) `The value of the 2 winner is 11.`
    `c:\>`

(b) `The value of the 3 winner is 9.`
    `c:\>`

(c) `The value of the 2 looser is 2.`
    `c:\>`

(d) `The value of the 3 looser is 2.`
    `c:\>`

(e) None of the above.

Question 6

Refer to Program A. What will you see if the following command is typed?

```
c:\>winnerloser l 2 3 8 4 7 9 11 2 1 33
```

(a) `The value of the 2 winner is 11.`
    `c:\>`

(b) `The value of the 3 winner is 9.`
    `c:\>`

(c) `The value of the 2 looser is 2.`
    `c:\>`

(d) `The value of the 3 looser is 2.`
    `c:\>`

(e) None of the above.

Question 7

Refer to Program B. If your option of sorting style is "d" the numbers entered are "4 5 2 1 6", what will you see below the numbers you entered?

(a) `The sorted numbers are 1 2 4 5 6.`
    `c:\>`

(b) `The sorted numbers are 1 2 4 5 6.`

    `c:\>`

(c) `The sorted numbers are 6 5 4 2 1.`
    `c:\>`

(d) `The sorted numbers are 6 5 4 2 1.`

    `c:\>`

(e) None of the above.

Question 8

Refer to Program B. If your option of sorting style is "a" the numbers entered are "4 5 2 1 6", what will you see below the numbers you entered?

(a) `The sorted numbers are 1 2 4 5 6.`
    `c:\>`

(b) `The sorted numbers are 1 2 4 5 6.`

    `c:\>`

(c) `The sorted numbers are 6 5 4 2 1.`
    `c:\>`

(d) `The sorted numbers are 6 5 4 2 1.`

    `c:\>`

(e) None of the above.

Question 9

Refer to Program C. Which number is assigned to "Spade A" during the initialization step?

(a) 0.

(b) 12.

(c) 24.

(d) 36.

(e) None of the above.

Question 10

Refer to Program C. Which of the following numbers are the possible outcomes of 'tmp1'?

(i) 0.

(ii) 1.

(iii) 52.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 11

Refer to Program C. How many integer variables are included in the integer array 'PLAYER'?

(a) 2.

(b) 5.

(c) 10.

(d) 52.

(e) None of the above.

Question 12

Refer to Program C. After executing the program 'poker02.exe', the following results are displayed on the screen.

```
Here are the card.
Player 1: S10 S11 S12 S13
Player 2: D10 D11 D12 D13

c:\>
```

If the program 'poker02.exe' is re-executed, what will you see?

(a) The same results.

(b)
```
Here are the card.
Player 1: S 1 S 2 S 3 S 4
Player 2: D 1 D 2 D 3 D 4

c:\>
```

(c)
```
Here are the card.
Player 1: C 1 C 2 C 3 C 4
Player 2: H 1 H 2 H 3 H 4

c:\>
```

(b) No one knows, as the cards are re-shuffled with a new set of random numbers.

Question 13

Refer to Program D. How many floating point number variables are included in 'S'?

(a) 10.

(b) 11.

(c) 100000.

(d) 100001.

(e) None of the above.

Question 14

Refer to Program D. Which of the following numbers are the possible outcomes of 'tmp'?

(i) 4.

(ii) 8.

(iii) 12.

3

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 15

Refer to Program D. Which one of the following functions is the function of 'nlfunction()'?

(a) $f(x) = \tanh\left(\frac{x}{2} - 1\right) - \frac{1}{10}$.

(b) $f(x) = \tanh\left(\frac{x-1}{2}\right) - \frac{1}{10}$.

(c) $f(x) = \tanh\left(\frac{x}{2} - 1\right) - 1$.

(d) $f(x) = \tanh\left(\frac{x-1}{2}\right) - 1$.

(e) None of the above.

Question 16

Refer to Program E. Which one of the following algorithms is the program being implemented?

(a) Initialization:
```
Input x (<0) and y (>2).
Input Precision error 'ERR'.

DO
{
tmp = (x+y)/2;
IF (f(tmp)*f(x) < 0)
    {
    y = tmp;
    DIFF = |f(x) - f(y)|;
    }
ELSE
    {
    x = tmp;
    DIFF = |f(x) - f(y)|;
    }
} WHILE (DIFF > ERR)
```

(b) Initialization:
```
Input x (<0) and y (>2).
Input Precision error 'ERR'.

DO
```

```
{
tmp = (x+y)/2;
IF (f(tmp)*f(x) < 0)
    {
    x = tmp;
    DIFF = |f(x) - f(y)|;
    }
ELSE
    {
    y = tmp;
    DIFF = |f(x) - f(y)|;
    }
} WHILE (DIFF > ERR)
```

(c) Initialization:
```
Input x (<0) and y (>2).
Input Precision error 'ERR'.

DO
{
tmp = (x+y)/2;
IF (f(tmp)*f(x) < 0)
    {
    y = tmp;
    DIFF = |f(x) - f(y)|;
    }
ELSE
    {
    x = tmp;
    DIFF = |f(x) - f(y)|;
    }
} WHILE (DIFF < ERR)
```

(d) Initialization:
```
Input x (<0) and y (>2).
Input Precision error 'ERR'.

DO
{
tmp = (x+y)/2;
IF (f(tmp)*f(x) < 0)
    {
    x = tmp;
    DIFF = |f(x) - f(y)|;
    }
ELSE
    {
    y = tmp;
    DIFF = |f(x) - f(y)|;
    }
} WHILE (DIFF < ERR)
```

(e) None of the above.

Question 17

```
#include<stdio.h>

main()
{
int MAX = 5;
int x[MAX], i;

printf("Input x0: ");
scanf("%d", &x[0]);
printf("%d", x[0]);
for(i = 1; i<MAX; i++)
{
x[i] = (x[i-1] + MAX)/MAX;
printf(" %d", x[i]);
}
}
```

Once the above program is executed, you will be asked to input the initial value for x_0. Then, the commands in the for-loop will generate x_1, x_2, x_3 and x_4 and display them on the screen. If the initial value of x_0 is 9, what will you see?

(a) ```
Input x0: 9
   2 1 1 1
   c:\>
```

(b) ```
Input x0: 9
   2 1 1 1

   c:\>
```

(c) ```
Input x0: 9
   9 2 1 1 1
   c:\>
```

(d) ```
Input x0: 9
   9 2 1 1 1

   c:\>
```

(e) None of the above.

Question 18

```
#include<stdio.h>

main()
{
int MAX = 5;
int x[MAX], i;

printf("Input x0: ");
scanf("%d", &x[0]);
printf("%d", x[0]);
```

```
for(i = 1; i<MAX; i++)
{
    x[i] = (x[i-1] + MAX)%MAX;
    printf(" %d", x[i]);
}
}
```

Once the above program is executed, you will be asked to input the initial value for x_0. Then, the commands in the for-loop will generate x_1, x_2, x_3 and x_4 and display them on the screen. If the initial value of x_0 is 9, what will you see?

(a) ```
Input x0: 9
   4 4 4 4
   c:\>
```

(b) ```
Input x0: 9
   4 4 4 4

   c:\>
```

(c) ```
Input x0: 9
   9 4 4 4 4
   c:\>
```

(d) ```
Input x0: 9
   9 4 4 4 4

   c:\>
```

(e) None of the above.

Question 19

```
#include<stdio.h>

main()
{
int MAX = 5, i;
float x[MAX];

printf("Input x0: ");
scanf("%f", &x[0]);
printf("%.4f", x[0]);
for(i = 1; i<MAX; i++)
{
x[i] = (x[i-1] + MAX)/MAX;
printf(" %.4f", x[i]);
}
}
```

Once the above program is executed and the input x_0 is 9.5, the line below the input is shown as follows :

```
9.5000 2.9000 1.5800 1.3160 1.2632
```

What will you see if the 'printf()' command inside the for-loop is changed to the following.

```
printf(" %.2f", x[i]);
```

(a) Input x0: 9.5
     9.50 2.90 1.58 1.31 1.26
     c:\>

(b) Input x0: 9.5
     9.50 2.90 1.58 1.31 1.26

     c:\>

(c) Input x0: 9.5
     9.50 2.90 1.58 1.32 1.26
     c:\>

(d) Input x0: 9.5
     9.50 2.90 1.58 1.32 1.26

     c:\>

(e) None of the above.

Question 20

```
#include<stdio.h>
#include<stdlib.h>
#include<time.h>

main()
{
  int MAX = 10, A[MAX], i;

  srand(time(NULL));
  printf("%d numbers are:\n", MAX);
  for(i=0; i<MAX; i++)
  {
    A[i] = rand() % 100 + 1;
    printf("%d ", A[i]);
  }
}
```

If 'stdio.h' library file is not included in the program, which one of the following functions will be reported '[**Error**] **functionname() was not declared in this scope.**'?

(a) time().

(b) srand().

(c) printf().

(d) rand().

Question 21

Refer to the program in Question 20. If 'time.h' instead of 'stdio.h' is not included, which one of the following functions will be reported '[**Error**] **functionname() was not declared in this scope.**'?

(a) time().

(b) srand().

(c) printf().

(d) rand().

Question 22

```
#include<stdio.h>
int f0(int), f1(int);

main()
{
int x=2, y;

while((x!=0)&&(x!=1))
    {
    printf("Input x (0 or 1): ");
    scanf("%d", &x);
    }
printf("Input y (any integer): ", y);
scanf("%d", &y);

if(x == 0)
    printf("Result: %d.", f0(y));
if(x == 1)
    printf("Result: %d.", f1(y));

}

int f0(int y)
{
int tmp;

tmp = y*y;

return tmp;
}

int f1(int y)
{
int tmp;

tmp = (y-1)*(y-1);

return tmp;
}
```

Which of the following function(s) is(are) the outcome of the program?

(i) $f(x, y) = (1 - x)y^2 + x(y - 1)^2$.

(ii) $f(x, y) = xy^2 + (1 - x)(y - 1)^2$.

(iii) $f(x, y) = x(1 - 2y) + y^2$.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 23

Which of the following program segments have either syntax or logical error?

(i)
```c
#include<stdio.h>

main()
{
    char  name[256];

    printf("Enter name: ");
    scanf("%s", &name);
    printf("Your name is %s.\n", name);
}
```

(ii)
```c
#include<stdio.h>

main()
{
    int  A, i = 0;

    A = 0;
    do
    {
      printf("%d ", A);
      A = A + i;
      i = i + 1;
    }while(i < 10);
}
```

(iii)
```c
#include<stdio.h>

main()
{
    int  A, i = 0;

    A = 0;
    for(i=0; i<10; i++);
    {
```

```c
      printf("%d ", A);
      A = A + i;
      i = i + 1;
    }
}
```

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 24

```c
/*********************
command_argument.cpp
********************/
#include<stdio.h>
#include<stdlib.h>

main(int argc, char *argv[])
{
    int i;

    for(i=0; i<argc; i++)
     printf("%s ", argv[i]);
}
```

If the following command is type on the command prompt, what will you see?

```
c:\>command_argument 1 3 2 6 4 5
```

(a) 3 2 6 4 5
    c:\>

(b) 1 3 2 6 4 5
    c:\>

(c) command_argument 3 2 6 4 5
    c:\>

(d) command_argument 1 3 2 6 4 5
    c:\>

(e) None of the above.

Question 25

Refer to the program in Question 24. What is the value of 'argc'?

(a) 5.

7

(b) 6.

(c) 7.

(d) 8.

Question 26

Which of the following software have a built-in interpreter?

(i) Dev C++

(ii) Matlab

(iii) Android

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 27

Which of the following statement about the interaction between an application software (AS) and the operating system (OS) is *False*?

(a) Once an AS has a request for an OS, a request message will then be sent to the service queue of the OS.

(b) Once the OS has completed the request, the result message will be sent to the receiving queue of the AS.

(c) Normally, an AS has no right to instruct the OS to change the priority of its request.

(d) Normally, an OS does not change the priority of any request from any AS.

(e) Normally, an OS will complete the requests in *first-in-first-out* (equivalent, *first-come-first-serve*) basis.

(f) None of them is false.

Question 28

How does an application software (AS) know that the request to the operating system (OS) has been completed?

(a) The AS from time to time will send a message to the OS checking for the progress.

(b) The AS from time to time will check from its queue seeing if the result message has been arrived.

(c) The OS from time to time will send a message to the AS informing the progress.

(d) The OS from time to time will check from its queue seeing if there is any *check-for-progress* message coming.

(e) None of the above.

Question 29

Which of the following statement about medium access control (MAC) is true?

(a) If a computer has detected that the medium is in use, it will wait for a fixed period of time and then re-detect the medium seeing if it is in use. If it is not, the computer sends out the message to the medium.

(b) If a computer has detected that the medium is in use, it will wait for a random period of time and then re-detect the medium seeing if it is in use. If it is not, the computer sends out the message to the medium.

(c) If a computer has detected that the medium is not in use, it will wait for a fixed period of time and then re-detect the medium seeing if it is in use. If it is not, the computer sends out the message to the medium.

(d) If a computer has detected that the medium is not in use, it will wait for a random period of time and then re-detect the medium seeing if it is in use. If it is not, the computer sends out the message to the medium.

Question 30

Which of the following statements about *port number* are true?

(i) Each application server (web server for instance) is usually assigned to at least one unique port number.

(ii) Once a message has arrived a computer, the network operating system will check the port number specified in the IP datagram and then pass the message to the corresponding application server queue.

(iii) Two different application servers, say email server and web server, can use the same port number.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 31

Which of the following stuffs are included in an IP datagram?

(i) Source and destination MAC addresses.

(ii) Source and destination IP addresses.

(iii) Source and destination PORT numbers.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 32

Which of the following statements about TTL are true?

(i) TTL refers to time-to-live.

(ii) Once an IP datagram has been received by a computer, it will check if the destination IP address is its IP address. If it is not and TTL is larger than 0, it will deduct the number in the TTL by 1 and then forward the IP datagram to a neighbor computer.

(iii) Once an IP datagram has been received by a computer, it will check if the destination IP address is its IP address. If it is not and TTL is 0, it will simply discard the IP datagram without forwarding the IP datagram to a neighbor computer.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 33

Which of the following factors are the causes of packet loss in wireless network?

(i) Background noise in the atmosphere, such as the sun radiation and other radio signals.

(ii) A packet has been forwarded for TTL times but still cannot arrive the destination IP address.

(iii) The computer of the destination IP address has been shut down.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 34

Which of the following factors will affect the data transmission rate in wireless network?

(i) Bit error rate.

(ii) Data signal frequency.

(iii) Carrier wave frequency.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 35

Which of the following protocols are not embraced in TCP/IP?

(i) HTTP.

(ii) FTP.

(iii) VOIP.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 36

Which of the following environments their medium access controls are essentially the same?

(i) Computer connecting in a ring structure.

(ii) Computers connecting in a bus topology.

(iii) Mobile devices and the access point.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 37

Suppose your cell phone (with enough battery power) has been set to the following conditions but your cell phone is not able to access www.yahoo.com. However, you can still make a phone call. Which of the following reason(s) is(are) likely the cause(s) of this connection failure?

```
WiFi              OFF
Bluetooth         OFF
Cellular          ON
Personal Hotspot  OFF
Carrier           ON
```

(i) No WiFi access point available.

(ii) The DNS in the telecom firm is down.

(iii) The www.yahoo.com web server is down.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 38

If we would like to show the following text on a webpage, which of the following codes should be added?

Adam Smith.

In the HTML file, what would be the correct coding for it?

(i) `<b><i>Adam Smith</i></b>`

(ii) `<i><b>Adam Smith</b></i>`

(iii) `<i><b>Adam Smith</i></b>`

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 39

Here is the file structure of the web files.

```
web/misc/poker01.cpp
        poker02.cpp
        bubblesort01.cpp
  /photo/john2009.jpg
         john2017.jpg
         john2019.jpg
  /index.html
   cs.html
```

Which of the following commands will function correctly if they are added in a file 'index.html'?

(i) `<a href="www.nchu.edu.tw">NCHU</a>`

(ii) `<a href="misc/john2009.jpg">John</a>`

(iii) `<a href="misc/poker01.cpp">Poker</a>`

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 40

Here is a segment of HTML code.

```
<table>
<tr> <td>Instructor:</td>
     <td>John Sum</td> </tr>
<tr> <td>Office:</td>
     <td>Room 821</td></tr>
<tr> <td>Time:</td>
     <td>Friday 09:10-12:00 </td> </tr>
<tr> <td>Classroom:</td>
     <td>CS121</td> </tr>
</table>
```

Once the above program segment is interpreted, what will you see on the browser.

(a) `Instructor: John Sum`
    `Office: Room 821`
    `Time: Friday 09:10-12:00`
    `Classroom: CS121`

(b) `Instructor:John Sum`
    `Office:Room 821`
    `Time:Friday 09:10-12:00`
    `Classroom:CS121`

(c) `Instructor:John Sum`
    `Office:    Room 821`
    `Time:      Friday 09:10-12:00`
    `Classroom: CS121`

(d) `Instructor: John Sum`
    `Office:     Room 821`
    `Time:       Friday 09:10-12:00`
    `Classroom:  CS121`

Question 41

Here is the access matrix for users John, Mary, Peter, application system A, and application system B.

	F1	F2	F3	A	B
John	orw	rw		e	e
Mary	r	w	orw	e	
Peter		orw	rw		e
A	rw	rw	rw		
B	rw	rw	rw		

o: owner, r: read, w: write, e: execute
EG: John can execute A and B to do anything.

Which of the following information leakage(s) could happen?

(i) Information in F1 could be leaked to Peter.

(ii) Information in F2 could be leaked to Mary.

(iii) Information in F3 could be leaked to John.

Answer:

(a) (i) and (ii) only.

(b) (ii) and (iii) only.

(c) (i) and (iii) only.

(d) (i), (ii) and (iii).

Question 42

Refer to the artificial CPU and its commands, what will be the content of $M4$ if the following commands are executed?

```
DEF M1 1
DEF M2 2
DEF M3 5

MOV IA M1
IF IA == 0
    MOV IA M2
    MOV IA M2
    MOV IB M3
    ADD IA IB
    MOV M4 OUT
ELSE
    MOV IA M1
    MOV IA M1
    MOV IB M3
    MUL IA IB
    MOV M4 OUT
ENDIF
```

(a) 2.

(b) 7.

(c) 5.

(d) 0.

Question 43

What will be the content of $M4$ if the following program segment is executed?

```
DEF M1 16
DEF M2 22
DEF M3 10
MOV IA M1
MOV IB M2
CMP IA IB
CMP IA IB
MOV M4 OUT
MOV IA M2
MOV IB M3
CMP IA IB
CMP IA IB
MOV IA OUT
MOV IB M4
ADD IA IB
MOV M4 OUT
```

(a) 28.

(b) 30.

(c) 32.

(d) 34.

Question 44

Refer to the artificial CPU and its commands, what will be the content of $M4$ if the following commands are executed?

```
DEF M1 0
DEF M2 2
DEF M3 5

MOV IA M1
IF IA == 0
   MOV IA M2
   SHL IA 00000100
   MOV IA OUT
   MOV IB M2
   ADD IA IB
   MOV M4 OUT
ELSE
   MOV IA M3
   SHL IA 00000100
   MOV IA OUT
   MOV IB M3
   ADD IA IB
   MOV M4 OUT
ENDIF
```

(a) 4.

(b) 6.

(c) 8.

(d) 10.

Question 45

Given that there are five memories M1, M2, M3, M4 and M5. Here is the program segment to instruct the circuit.

```
----------------
MOV IA M1
MOV IB M2
MUL IA IB
MOV IA OUT
MOV IB M3
MUL IA IB
MOV IA OUT
MOV IB M4
SUB IA IB
MOV M5 OUT
----------------
```

which of the following mathematical equation is identical to the operation of the following program segment?

(a) $M5 = M4 - M1 \times M2 \times M3$.

(b) $M5 = M4 - (M1 + M2) \times M3$

(c) $M5 = M1 \times M2 \times M3 - M4$.

(d) $M5 = (M1 + M2) \times M3 - M4$.

Diagram for Questions 46-50

The following schematic diagram is for Question 11 to Question 15. It is a circuit consisting of two logic gates.
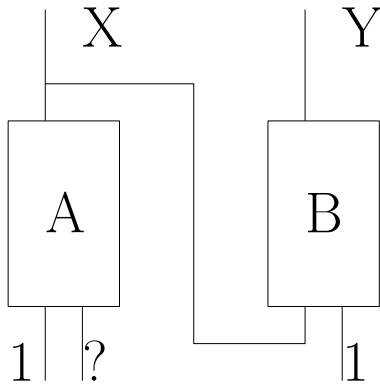
Question 46

What are the output values X and Y if A is an XOR gate, B is an AND gate and the input (from left to right) is 101.

(a) $X = 0, Y = 0$.

(b) $X = 0, Y = 1$.

(c) $X = 1, Y = 0$.

(d) $X = 1, Y = 1$.

Question 47

What are the output values X and Y if A is an OR gate, B is an OR gate and the input (from left to right) is 111.

(a) $X = 0, Y = 0$.

(b) $X = 0, Y = 1$.

(c) $X = 1, Y = 0$.

(d) $X = 1, Y = 1$.

Question 48

What are the output values X and Y if A is an AND gate, B is an XOR gate and the input (from left to right) is 101.

(a) $X = 0, Y = 0$.

(b) $X = 0, Y = 1$.

(c) $X = 1, Y = 0$.

(d) $X = 1, Y = 1$.

Question 49

What are the output values X and Y if A is an OR gate, B is an NAND gate and the input (from left to right) is 111.

(a) $X = 0, Y = 0$.

(b) $X = 0, Y = 1$.

(c) $X = 1, Y = 0$.

(d) $X = 1, Y = 1$.

Question 50

What are the output values X and Y if A is an NAND gate and B is an NAND gate and the input (from left to right) is 111.

(a) $X = 0, Y = 0$.

(b) $X = 0, Y = 1$.

(c) $X = 1, Y = 0$.

(d) $X = 1, Y = 1$.

APPENDIX

In this appendix, it includes the information about the artificial CPU and program source codes. Please read them carefully!

Artificial CPU

Below is a simple circuit. It consists of a **memory** with 16 memory spaces (from M1 to M16), an **ALU block**, 2 **input registers** (IA and IB) and one **output register** (OUT). M1 to M16, IA, IB and OUT are all 8 bits long. Numbers are represented in *2's compliment* format.
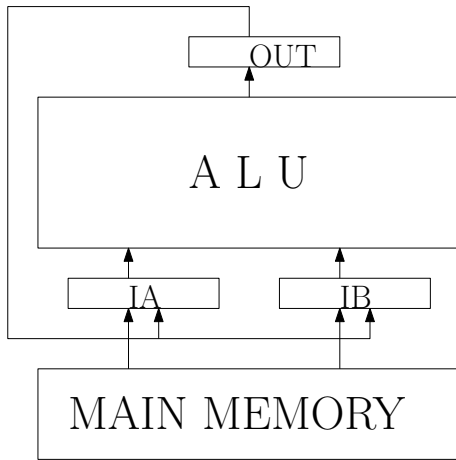
Eleven commands (MOV, ADD, SUB, MUL, DIV, CMP, SHL, SHR, DEF, MSK and IF) are provided for instructing the above circuit. The syntax and the descriptions of these commands are depicted in Table 1.

Notes on CPU Commands

1. For the "CMP" command, if $X = 0110$ and $Y = 1101$, $OUT = 1011$.

2. For "SHL" and "SHR" commands, the content of Y can only be one of the following.

Y	Meaning
10000000	(Shift 7 bits)
01000000	(Shift 6 bits)
00100000	(Shift 5 bits)
00010000	(Shift 4 bits)
00001000	(Shift 3 bits)
00000100	(Shift 2 bits)
00000010	(Shift 1 bits)
00000001	(No shift)

Table 1: Commands for using the CPU.

Syntax	Description
MOV X Y	Copy the content of Y to X
ADD X Y	$OUT = X + Y$.
SUB X Y	$OUT = X - Y$.
MUL X Y	$OUT = X \times Y$.
DIV X Y	$OUT = X/Y$.
CMP X Y	$OUT = b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8$. $b_i = 0$ if $X_i = Y_i$. $b_i = 1$ if $X_i \neq Y_i$.
SHL X Y	OUT is the content of X shifting left Y bits.
SHR X Y	OUT is the content of X shifting right Y bits.
DEF X N	Define X as the number N.
MSK X M	Mask the value of X by M.
IF ELSE	Condition statement.

For example, if

$$X = 00011000, Y = 00000100,$$

the OUT of "SHL X Y" is 01100000 and the OUT of "SHR X Y" is 00000110.

3. For the "DEF" command, N must be a number in *decimal* form. X can only be a memory location. "DEF" command is not applicable for assigning values to a register. It is used to assign a value to a memory location. For example, "DEF M1 12" means that memory location $M1$ will be assigned with a value 12. Therefore, $M1 = 00001100$.

4. For the "MSK" command, it is used for masking a register (either IA or IB) by the mask M (in binary). The mask must be 8 bits long.

Suppose that the content of IA and M are defined as follows :

$$IA = 01001001, M = 11110000.$$

Then, the output OUT will be "01000000". The last four bits are masked. Here is an example.

```
------------------------
DEF M1 45
```

```
MOV IA M1
MSK IA 00001111
MOV M2 OUT
------------------------
```

Initially, $M1$ is assigned with value 45. In binary form, the content reads "00101101". Thus, the output OUT is "00001101".

5. The "IF-ELSE" command is an advanced level command. It is for conditional statement. Once it is executed, the CPU will performs multiple steps in order to make it works. You do not need to know the detail how it works. In term of its usage, it is simple. Here is an example.

```
------------------------
DEF M1 1
DEF M2 2
DEF M3 1

MOV IA M1
IF IA == 0
    MOV IA M2
    MOV IB M3
    ADD IA IB
    MOV M4 OUT
ELSE
    MOV IA M1
    ADD IB M2
    MOV M4 OUT
ENDIF
------------------------
```

Command "IF" checks if the content of IA is identical to "0". If it is, it will perform $M2+M3$ and output the result to $M4$. Otherwise, it will perform $M1+M2$ and output the result to $M4$.

```
------------------------
DEF M1 1
DEF M2 2
DEF M3 1

MOV IA M1
IF IA == 0
    MOV IA M2
    MOV IB M3
    ADD IA IB
```

```
    MOV M4 OUT
ENDIF
------------------------
```

In this example, the CPU performs $M2 + M3$ only if IA is zero. Otherwise, it performs nothing.

6. For the "IF-ELSE" command, the following conditions are allowed for you to define. Here NUM must be stated in decimal form but not in binary.

```
------------------------
IA == NUM
IA > NUM
IA >= NUM
IA < NUM
IA <= NUM
------------------------
```

Program A

The filename of the following program is "winnerloser.cpp".

```cpp
/***************************************************************************
winnerloser.cpp
***************************************************************************/
#include<stdio.h>
#include<stdlib.h>

main(int argc, char *argv[])
{
    int A[256]; /* Define integer array. */
    int i,j; /* Define indices. */
    int tmp; /* Define dummy variable for sorting. */
    char Opt = 'c';
    int SNUM, POS;

    /* Step 1: Read input from the command line. */
    SNUM = argc - 3;
    Opt = argv[1][0];
    POS = atoi(argv[2]);

    if (POS <= SNUM)
    {
      if((Opt == 'w')||(Opt == 'l'))
      {
        for(i = 0; i<SNUM; i++)
            A[i] = atoi(argv[i+3]);

        /* Step 2: Sorting the numbers */
        if(Opt == 'l')
        {
          for(i=0; i<SNUM-1; i++)
          for(j=0; j<SNUM-i-1; j++)
          {
                if(A[j] > A[j+1])
                {
                tmp = A[j];
                A[j] = A[j+1];
                A[j+1] = tmp;
                }
          }
          printf("The value of the %d loser is %d.", POS, A[POS-1]);
        }
        if(Opt == 'w')
        {
          for(i=0; i<SNUM-1; i++)
          for(j=0; j<SNUM-i-1; j++)
          {
                if(A[j] < A[j+1])
                {
                tmp = A[j];
                A[j] = A[j+1];
                A[j+1] = tmp;
                }
          }
          printf("The value of the %d winner is %d.", POS, A[POS-1]);
        }
```

```c
    }
    else
        printf("Something wrong in your inputs.");
    }
    else
        printf("The position %d is larger than %d.", POS, SNUM);
}
```

Program B

```c
#include<stdio.h>
main()
{
    int A[5], i, j, tmp;
    char OPT[8], Opt = 'c';

    printf("This program demonstrates how to do sorting.\n\n");
    while((Opt != 'a')&&(Opt != 'd'))
    {
    printf("Select a sorting style, 'a' for ascending and 'd' for descending.\n");
    printf("Option : ");
    scanf("%s", OPT);
    Opt = OPT[0];
    }
    printf("\n");
    printf("Please enter five numbers.\n");

    for(i=0; i<5; i++)
    {
    printf("Enter the %d number: ", i+1);
    scanf("%d", &A[i]);
    }

    if(Opt == 'a')
    {
        for(i=0; i<4; i++)
        for(j=0; j<4-i; j++)
                if(A[j] > A[j+1])
                {
                tmp = A[j];
                A[j] = A[j+1];
                A[j+1] = tmp;
                }
        printf("The sorted numbers are");
        for(i=0; i<5; i++)
            printf(" %d", A[i]);
        printf(".");
    }
    else
    {
        for(i=0; i<4; i++)
        for(j=0; j<4-i; j++)
                if(A[j] < A[j+1])
                {
                tmp = A[j];
                A[j] = A[j+1];
                A[j+1] = tmp;
                }
        printf("The sorted numbers are");
        for(i=0; i<5; i++)
            printf(" %d", A[i]);
        printf(".");
    }
}
```

Program C

The filename of the following program is "poker02.cpp".

```
/***************************************************
poker02.cpp
***************************************************/
#include<stdio.h>
#include<stdlib.h>
#include<time.h>

main()
{
        int MAX = 10000, RANGE = 52;
        int CARD[52], PLAYER[2][5], HAND, NUMBER;
        int i, j, tmp, tmp1, tmp2;

        /* Step 1: Card Initialization */
        for(i=0; i<RANGE; i++)
          CARD[i] = i;

        /* Step 2: Card shuffling */
        printf("The card are being shuffled now.\n");
        for(i=0; i<MAX; i++)
        {
                tmp1 = rand() % RANGE;
                tmp2 = rand() % RANGE;
                tmp = CARD[tmp1];
                CARD[tmp1] = CARD[tmp2];
                CARD[tmp2] = tmp;
        }

        /* Step 3: Card Assignment */
        for(i=0; i<2; i++)
          for(j=0; j<5; j++)
            PLAYER[i][j] = CARD[i*4+j];

        printf("Here are the cards.\n");
        for(i=0; i<2; i++)
        {
            printf("Player %2d: ", i+1);
            for(j=0; j<5; j++)
              {
                tmp = PLAYER[i][j];
                HAND = tmp/13;
                NUMBER = tmp % 13;

                switch ( HAND )
                {
                        case 0: printf("S%2d   ", NUMBER+1); break;
                        case 1: printf("H%2d   ", NUMBER+1); break;
                        case 2: printf("C%2d   ", NUMBER+1); break;
                        case 3: printf("D%2d   ", NUMBER+1); break;
                }
              }
            printf("\n");
        }
        /* END */
}
```

Program D

```
/**************************************************************************
random_number_03.cpp

Description: This program shows the distribution of z = (x + y),
    where x and y are uniform distributed random variables in
    {0, 1, 2, 3, 4, 5}. So, there are 11 possible outcomes of z.
    z = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10}.


**************************************************************************/
#include<stdio.h>
#include<stdlib.h>
#include<time.h>

main()
{
        int MAX = 100000;
        int RANGE = 6;
        float S[2*RANGE-1];
        int i, tmp, tmp1, tmp2;

        srand(time(NULL));
        for(i=0; i<2*RANGE-1; i++)
                S[i] = 0;

        printf("%d random numbers from [1:%d] are generated.\n", MAX, RANGE);
        for(i=0; i<MAX; i++)
        {
                tmp1 = rand() % RANGE;
                tmp2 = rand() % RANGE;
                tmp = tmp1 + tmp2;
                S[tmp] = S[tmp] + 1;
        }

        printf("\n");
        printf("The statistics of the random numbers are listed below.\n");
        for(i=0; i<2*RANGE-1; i++)
        {
          printf("Proportion of %d is %0.4f. (%4.0f %d)\n", i, S[i]/MAX, S[i], MAX);
        }
}
```

Program E

```
/************************************************************
bisection01.cpp
************************************************************/
#include<stdio.h>
#include<math.h>

float nlfunction(float xtmp);

main()
{
    float ERR; /* Error for stopping criteria */
    float a; /* Step size */
    float x, y, tmp;    /* variable for iteration */
    float fx, fy, ft;
    float precision;
    int k = 0;

    printf("Finding the root of tanh(0.5(x-1)) - 0.1.\n");
    printf("Bisection method is applied.\n");
    printf("Enter the initial guess of x (< 0): ");
    scanf("%f", &x);
    printf("Enter the initial guess of y (> 2): ");
    scanf("%f", &y);
    printf("Precision: ");
    scanf("%f", &precision);

    /* Bisection Method */
    ERR = y - x;
    fx = nlfunction(x);
    fy = nlfunction(y);
    while((ERR > precision)&&(fx*fy < 0))
    {
     tmp = (x + y)/2;
     ft = nlfunction(tmp);
     if(fx*ft < 0)
{
y = tmp;
fy = ft;
}
if(ft*fy < 0)
{
x = tmp;
fx = ft;
}
     ERR = y - x;
     k = k + 1;
     printf("Step %d: [%0.8f, %0.8f]\n", k, x, y);
    }

    /* Output the results on screen */
    printf("The root of f(x) is at %0.8f.\n", x);
}

float nlfunction(float xtmp)
{
float ftmp;
```

21

```
ftmp = tanh(0.5*(xtmp-1)) - 0.1;

return ftmp;
}
```