國立成功大學

112學年度碩士班招生考試試題

編　號：196

系　所：製造資訊與系統研究所

科　目：計算機概論

日　期：0206

節　次：第 2 節

備　註：不可使用計算機

※ 考生請注意：本試題不可使用計算機。 請於答案卷(卡)作答，於本試題紙上作答者，不予計分。

A. True or False Questions. (50%)

1. Descriptions of **Computer System and Architecture. (10%)**

_____(1) Operating system is a collection of software that manages the hardware and software resources of a computer. Operating system is responsible for managing the computer's memory, processing power, and input/output devices. However, it does not handle tasks such as connecting to networks and managing files.

_____(2) Each operating system has its own unique features and capabilities, and is designed to work with specific types of hardware and software. Some examples of popular operating systems include Microsoft Office, Linux, Android, iOS, Line.

_____(3) Distributed systems are designed to be small, lightweight, and low-power, and they often operate in real-time, meaning that they must respond to events within a certain time frame. They are often found in devices that require real-time processing, such as control systems and devices that require precise timing.

_____(4) Embedded systems are often found in devices that we use every day, such as smartphones, cars, appliances, and industrial control systems. The software in an embedded system is often called firmware, and it is stored in the system's non-volatile memory.

_____(5) An Instruction Set Architecture (ISA) is part of the abstract model of a computer that defines how the CPU is controlled by the software. The ISA acts as an interface between the hardware and the software, specifying both what the processor is capable of doing as well as how it gets done.

2. Descriptions of **Computer hardware. (10%)**

_____(1) Computer hardware refers to the physical components of a computer, such as the processor, memory, storage, and input/output devices. In addition to these components, a computer may also include other hardware such as a motherboard, power supply, and various ports and connectors for connecting to other devices.

_____(2) Processor: Also known as the central processing unit (CPU), the processor is the brain of the computer and where the computer stores data that it is currently using or processing. A processor only understands instructions encoded in some numerical fashion, usually as binary numbers.

_____(3) Memory: Memory, or RAM (random access memory). It executes instructions and performs calculations for the computer.

_____(4) Storage: Storage devices, such as hard drives and solid-state drives, are where the computer stores long-term data, such as the operating system, software applications, and user files.

_____(5) Input/output devices: These are the devices that allow a user to interact with the computer, such as a keyboard, mouse, and monitor. Other examples of input/output devices include printers, scanners, and speakers.

3. Descriptions of **Data. (10%)**

_____(1) Data processing is the process of collecting, storing, and organizing data, and then using that data to produce useful information.

_____(2) The collected data is stored in a database or other storage system. There are many types of

databases, including relational databases, object-oriented databases, and NoSQL databases. All databases have the same features and are suited to all types of data and applications.

_____(3) Object-oriented databases are a type of database that is based on the concept of tables, which are made up of rows and columns, while relational databases are designed to support the storage and management of complex data structures and to provide better integration with specific programming languages.

_____(4) Data fusion is the process of combining data from multiple sources to produce a more complete, accurate, and useful representation of a system or phenomenon. Common approaches of data fusion include filtering, feature extraction, data association, data integration, and decision making.

4. Descriptions of **Programming language. (10%)**

_____(1) A programming language is a set of instructions, rules, and conventions that are used to create software programs and applications.

_____(2) In computing, a compiler is a computer program that translates computer code written in one programming language (the source language) into another language (the target language).

_____(3) Code editors are programming language-specific. Some editors support one or two programming languages whereas some support multiple programming languages. It can give suggestions and highlights based on language support.

_____(4) Assembly language is a low-level programming language with a very strong correspondence between the instructions in the language and the architecture's machine code instructions.

_____(5) High level programming languages include BASIC, C/C++, Java and ASCII code.

5. Descriptions of **Applications and AI. (10%)**

_____(1) An application program (software application, or application, or app for short) is a computer program designed to carry out a specific task other than one relating to the operation of the computer itself, typically to be used by end-users.

_____(2) Android and iOS use different programming languages for their operating systems. Android apps are written in Objective-C/Swift, whereas iOS apps are written in Java.

_____(3) An integrated development environment (IDE) is a software application that provides comprehensive facilities to computer programmers for software development. An IDE normally consists of at least a source code editor, build automation tools and a debugger. Some IDEs, such as NetBeans and Eclipse, contain the necessary compiler, interpreter, or both; others, such as SharpDevelop and Lazarus.

_____(4) TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks. TensorFlow can only be used in Python.

_____(5) Supervised learning algorithms take a set of data that contains only inputs, and find structure in the data, like grouping or clustering of data points. The algorithms, therefore, learn from test data that has not been labeled, classified or categorized.

B. Blank Filling Questions-Networks. (10%)

Please fill in the correct terms according to the description.

Repeaters, hubs, switches, routers, modems, firewalls, multiprocessing, multithreading.

1. _____ An internetworking device that forwards packets between networks by processing the addressing or routing information included in the packet. The routing information is often processed in conjunction with the routing table.
2. _____ A network software or device for controlling network security and access rules. They are inserted in connections between secure internal networks and potentially insecure external networks such as the Internet.
3. _____ It is supported by the operating system that make a central processing unit (CPU) to provide multiple threads of execution concurrently. The resources of a single or multiple cores are shared, including the computing units, the CPU caches, and the translation lookaside buffer (TLB).
4. _____ A network device forwards frames to the ports involved in the communication based on the destination MAC address in each frame. It learns the association of physical ports to MAC addresses by examining the source addresses of received frames and only forward the frame when necessary.
5. _____ The use of two or more central processing units (CPUs) within a single computer system. The term also refers to the ability of a system to support more than one processor or the ability to allocate tasks between them.

C. Blank Filling Questions-Performance metrics. (10%)

Please fill in the correct terms according to the description.

Accuracy, Mean Absolute Error, Precision, Root Mean Squared Error, Mean Squared Error, F1-score, Recall.

1. _____ $\frac{1}{N}\sum_{j=1}^{N}(y_j - \tilde{y}_j)^2$
2. _____ $\frac{1}{N}\sum_{j=1}^{N}|y_j - \tilde{y}_j|$
3. _____ $\sqrt{\frac{1}{N}\sum_{j=1}^{N}(y_j - \tilde{y}_j)^2}$
4. _____ the ratio of true positives and total positives predicted
5. _____ the number of correct predictions divided by the total number of predictions, multiplied by 100.

D. Blank Filling Questions-Algorithms. (10%)

There are many different sorting algorithms that can be used to rearrange a list of items in a particular order. Some common sorting algorithms include: *Quick sort, Bubble sort, Insertion sort, Selection sort, Merge sort* and *Heap sort.* Each of these algorithms has its own advantages and disadvantages, and the best choice for a particular situation will depend on the specific requirements and constraints of the problem. Please fill in the correct sorting algorithm according to the description.

1. _____ This is a simple sorting algorithm that repeatedly compares adjacent elements and swaps them if they are in the wrong order.
2. _____ This algorithm repeatedly selects the smallest element in the list and moves it to the front of the list.
3. _____ This algorithm works by dividing the list into smaller sublists, sorting those sublists, and then merging them back together.
4. _____ This algorithm uses a pivot element to partition the list and then recursively sorts the sublists on either side of the pivot.
5. _____ This algorithm works by repeatedly inserting elements into the correct position in a sorted list.

E. Blank Filling Questions-Codes. (20%)

There are many different programming languages, each with its own syntax, semantics, and characteristics. Some popular programming languages include C, C++, Java, Python, and JavaScript. Please identify what the programming languages of the following codes in figure 1 to figure 4 and also answer the output of each codes.

	Programming language	Output of the code
Figure 1	(1) (2%)	(2) (3%)
Figure 2	(3) (2%)	(4) (3%)
Figure 3	(5) (2%)	(6) (3%)
Figure 4	(7) (2%)	(8) (3%)

國立成功大學 112 學年度碩士班招生考試試題

系　　所：製造資訊與系統研究所

考試科目：計算機概論

考試日期：0206，節次：2

Figure 1

```
#include <math.h>
#include <stdio.h>
int main()
{
        int n, i, flag = 1;
        n = 13;
        for (i = 2; i <= sqrt(n); i++) {
                if (n % i == 0) {
                        flag = 0;
                        break;
                }
        }
        if (n <= 1)
                flag = 0;
        if (flag == 1) {
                printf("Yes");
        }
        else {
                printf(  "No");
        }
        return 0;
}
```

Figure 2

```
#include<iostream>
using namespace std;
void findFact(int, int *);
int main()
{
 int num =4, fact=1;
 findFact(num, &fact);
 cout<<"\nFactorial = "<<fact;
 cout<<endl;
 return 0;
}
void findFact(int n, int *f)
{
 int i;
 for(i=n; i>=1; i--)
 *f = (*f)*i;
}
```

Figure 3

```
let hcf;
const number1 = 6;
const number2 = 8;
for (let i = 1; i <= number1 && i <= number2; i++) {
if( number1 % i == 0 && number2 % i == 0) {
      hcf = i;
   }
}
let lcm = (number1 * number2) / hcf;
console.log(`Answer is ${lcm}.`);
```

Figure 4

```
lower = 10
upper = 15
for num in range(lower, upper + 1):
    if num > 1:
        for i in range(2, num):
            if (num % i) == 0:
                break
        else:
            print(num)
```