Year 9 Computer Science Exam Revision Guide

Autumn Term: Algorithms & Programming

Python Programming - Key Concepts

Variables and Data Types:

- **String:** Text data (e.g., "Hello")
- **Integer:** Whole numbers (e.g., 42)
- **Float:** Decimal numbers (e.g., 3.14)
- **Boolean:** True/False values

Programming Constructs:

- **Selection:** if/elif/else statements for decision making
- **Iteration:** for loops (count controlled) and while loops (condition controlled)

Example:

```
age = int(input("Enter your age: "))
if age >= 18:
    print("You can vote")
else:
    print("Too young to vote")
```

Data Handling:

- **Operators:** <, >, <=, >=, ==, !=
- **Boolean conditions:** and, or, not
- **Input validation:** Checking data is correct type/range

Algorithms:

- **Abstraction:** Breaking down complex problems
- **Decomposition:** Splitting problems into smaller parts
- **Subroutines:** Functions that can be called multiple times

YEAR 9 COMPUTER SCIENCE - Algorithms & Programming

Autumn Term Revision Guide

VARIABLES & DATA TYPES

STRING
Text data
"Hello World"

INTEGER
Whole numbers
42, -10, 0

FLOAT
Decimal numbers
3.14, 2.5

BOOLEAN
True/False values
True, False

PROGRAMMING CONSTRUCTS

SELECTION
Decision making

```
if age >= 18:
print("Adult")
else: print("Child")
```

ITERATION
Repeating code

```
for i in range(5):
print(i)
while x < 10:
```

OPERATORS & CONDITIONS

COMPARISON OPERATORS

- < (less than)
- > (greater than)
- <= (less than or equal)
- == (equal to)

BOOLEAN OPERATORS

- and (both conditions true)
- or (one condition true)
- not (opposite of condition)

Example: age > 16 and age < 65

CODE EXAMPLE

```
# Get user input
age = int(input("Enter your age: "))
# Check voting eligibility
if age >= 18:
   print("You can vote")
else:
   print("Too young to vote")
```

ALGORITHM DESIGN

ABSTRACTION
Hide unnecessary details, focus on main problem

DECOMPOSITION
Break complex problems into smaller parts

SUBROUTINES
Functions that can be called multiple times

INPUT VALIDATION

Checking Data is Correct

- Type check: Is it a number when expected?
- Range check: Is age between 0 and 120?
- Presence check: Has user entered something?
- Length check: Password at least 8 characters?

LOOP TYPES

FOR LOOP
Count controlled - repeats a specific number of times

```
for i in range(10):
```

WHILE LOOP
Condition controlled - repeats while condition is true

```
while x < 10:
```

EXAM TIPS

Key Skills for Programming Questions:

- ✓ Trace through code step by step
- ✓ Identify data types from examples
- ✓ Explain what selection and iteration do
- ✓ Write simple if statements and loops

COMMON MISTAKES TO AVOID

✗ Confusing = (assignment) with == (comparison) ✗ Forgetting indentation in Python ✗ Using wrong data type
✗ Infinite loops (while condition never becomes false) ✗ Not validating user input ✗ Mixing up and/or operators

Spring Term: Computer Systems

Fetch-Decode-Execute (FDE) Cycle

1. **Fetch:** CPU gets instruction from memory
2. **Decode:** CPU works out what instruction means
3. **Execute:** CPU carries out the instruction

Key Components:

- **CPU:** Central Processing Unit - brain of computer
- **RAM:** Random Access Memory - temporary storage
- **ROM:** Read Only Memory - permanent storage
- **Cache:** Very fast temporary storage

Hardware vs Software

Hardware: Physical components you can touch

- CPU, RAM, Hard drive, Motherboard

Software: Programs and instructions

- **System software:** Operating system, drivers
- **Application software:** Games, word processors
- **Utility software:** Antivirus, file managers

Storage Types

Primary Storage:

- **RAM:** Fast, volatile (loses data when power off)
- **ROM:** Slower, non-volatile (keeps data when power off)

Secondary Storage:

- **Magnetic:** Hard disk drives (HDD) - cheap, large capacity
- **Optical:** CDs, DVDs - portable, medium capacity
- **Solid State:** SSDs, USB drives - fast, reliable, expensive

Embedded vs Non-Embedded Systems

Embedded: Built into other devices (washing machine, car) **Non-Embedded:** General purpose computers (PC, laptop)

YEAR 9 COMPUTER SCIENCE - Computer Systems

Spring Term Revision Guide

FETCH-DECODE-EXECUTE (FDE) CYCLE

1. FETCH
CPU gets instruction from memory
Retrieve next instruction

2. DECODE
CPU works out what instruction means
Interpret instruction

3. EXECUTE
CPU carries out the instruction
Perform action

CYCLE REPEATS MILLIONS OF TIMES PER SECOND

KEY COMPUTER COMPONENTS

CPU
Central Processing Unit
Brain of computer

RAM
Random Access Memory
Temporary storage

ROM
Read Only Memory
Permanent storage

CACHE
Very fast storage
Speed boost

HARDWARE vs SOFTWARE

HARDWARE
Physical components you can touch
- CPU (processor)
- RAM (memory)
- Hard drive
- Motherboard

SOFTWARE
Programs and instructions
- System: OS, drivers
- Application: Games, apps
- Utility: Antivirus
- File managers

STORAGE TYPES

PRIMARY STORAGE
RAM: Fast, volatile (loses data)
ROM: Slower, non-volatile (keeps da

SECONDARY STORAGE
Magnetic (HDD):
Cheap, large capacity, slower
Optical (CD/DVD):
Portable, medium capacity
Solid State (SSD):
Fast, reliable, expensive

STORAGE COMPARISON

TYPE	SPEED	CAPACITY	COST
RAM	Very Fast	Medium	High
ROM	Medium	Small	Medium
HDD	Slow	Very Large	Low
SSD	Fast	Large	High
CD/DVD	Slow	Small	Very Low

EMBEDDED vs NON-EMBEDDED SYSTEMS

EMBEDDED
Built into other devices
- Car computers
- Washing machines
- Smart TVs

NON-EMBEDDED
General purpose computers
- Desktop PC
- Laptop
- Smartphone

EXAM TIPS

Key Skills for Computer Systems Questions:

- ✓ Explain each step of FDE cycle in correct order
- ✓ Compare storage types by speed, capacity, cost, volatility
- ✓ Give examples of embedded vs non-embedded systems
- ✓ Distinguish between hardware and software with examples
- ✓ Know when to use different storage types
- ✓ Understand role of CPU, RAM, ROM, Cache

COMMON MISTAKES TO AVOID

✗ Confusing RAM and ROM ✗ Wrong FDE cycle order ✗ Mixing up volatile/non-volatile ✗ Not explaining why storage chosen

Summer Term: Networking

Network Types

LAN (Local Area Network): Small area (home, school) **WAN (Wide Area Network):** Large area (internet, country)

Connection Types:

- **Wired:** Ethernet cables - faster, more reliable
- **Wireless:** WiFi, Bluetooth - convenient, can be slower

Network Topologies

Bus Topology:

- All devices connected to one main cable
- Cheap but if main cable fails, whole network fails

Star Topology:

- All devices connected to central hub/switch
- More expensive but more reliable

Network Security

Malware Types:

- **Virus:** Attaches to files, spreads when file shared
- **Worm:** Spreads automatically through networks
- **Trojan:** Disguised as legitimate software
- **Spyware:** Secretly monitors user activity

Security Measures:

- **Passwords:** Use strong, unique passwords
- **Antivirus software:** Detects and removes malware
- **Firewalls:** Block unauthorized network access
- **Encryption:** Scrambles data to make it unreadable

Animation Basics

Stop Frame Animation: Taking photos of objects in slightly different positions **Frame Rate:** Number of frames per second (fps) **File Formats:** JPEG (photos), PNG (images with transparency), GIF (simple animations)

YEAR 9 COMPUTER SCIENCE - Networking

Summer Term Revision Guide

NETWORK TYPES

LAN
Local Area Network

Small area coverage

🏠 Home networks

🏫 School networks

WAN
Wide Area Network

Large area coverage

🌐 Internet

🌍 Country networks

CONNECTION TYPES

WIRED
Ethernet cables

✓ Faster

✓ More reliable

✗ Less convenient

WIRELESS
WiFi, Bluetooth

✓ Convenient

✓ Mobile

✗ Can be slower

NETWORK TOPOLOGIES

BUS TOPOLOGY

All devices on one main cable

✓ Cheap to set up

✗ If cable fails, all fail

✗ Performance degrades

STAR TOPOLOGY

All devices connect to hub

✓ More reliable

✗ More expensive

MALWARE TYPES

VIRUS
Attaches to files

Spreads when file is shared

WORM
Spreads automatically through networks without files

TROJAN
Disguised as legitimate software

SPYWARE
Secretly monitors user activity

SECURITY MEASURES

PASSWORDS
🔐 Strong & unique

Mix of letters, numbers, symbols

8+ characters

ANTIVIRUS
🛡️ Detects & removes malware

Regular updates needed

FIREWALL
🔥 Blocks unauthorized network access

Filters incoming/ outgoing traffic

ENCRYPTION
🔒 Scrambles data

Makes it unreadable

ANIMATION BASICS

STOP FRAME
📸 Take photos of objects in slightly different positions

Play back quickly

FRAME RATE
🎬 Frames per second (fps)

24 fps = smooth

12 fps = choppy

FILE FORMATS
JPEG: Photos | PNG: Transparency

GIF: Simple animations

EXAM TIPS

Key Skills for Networking Questions:

- ✓ Compare LAN vs WAN with examples and coverage area
- ✓ Explain advantages/disadvantages of topologies
- ✓ Match security threats with appropriate countermeasures
- ✓ Identify malware types from descriptions
- ✓ Choose wired vs wireless for given scenarios
- ✓ Explain how animation frame rate affects quality

COMMON MISTAKES TO AVOID

✗ Confusing virus and worm ✗ Wrong topology advantages ✗ Mixing up LAN/WAN coverage ✗ Not matching security to threat

Key Exam Skills

Programming Questions

- **Trace through code:** Follow variables step by step
- **Identify errors:** Syntax errors (spelling) vs Logic errors (wrong result)
- **Write algorithms:** Use sequence, selection, iteration

System Questions

- **Compare storage types:** Speed, capacity, cost, volatility
- **Explain FDE cycle:** Know the three steps and what happens
- **Hardware vs Software:** Identify and classify components

Network Questions

- **Compare topologies:** Advantages and disadvantages
- **Security threats:** Identify malware types and countermeasures
- **Network performance:** Factors affecting speed and reliability

---

Common Exam Question Types

1. Code Analysis (4-6 marks)

- Given a Python program, trace through and predict output
- Identify what the program does
- Spot and correct errors

2. System Components (6-8 marks)

- Explain role of CPU, RAM, storage
- Compare different storage types
- Describe FDE cycle

3. Network Design (4-6 marks)

- Choose appropriate topology for given scenario
- Explain security measures needed
- Compare wired vs wireless

4. Problem Solving (8-10 marks)

- Design algorithm for given problem
- Use flowchart or pseudocode
- Include selection and iteration

YEAR 9 COMPUTER SCIENCE - Common Exam Questions & Answers

Essential Q&A for Exam Success

PROGRAMMING QUESTIONS

CODE TRACE (4-6 marks)

```
x = 5
y = 3
if x > y:
  print(x + y)
else:
  print(x - y)
```

ANSWER: Outputs 8
(5 > 3 is True, so 5 + 3 = 8)

ERROR TYPES (2-3 marks)

SYNTAX ERROR:

```
prin("Hello") # Missing 't'
```

LOGIC ERROR:

```
total = 0
total = total * 2 # Should be +
```

Syntax = won't run | Logic = wrong result

SYSTEM QUESTIONS

STORAGE TYPES (6-8 marks)

Q: Compare RAM and HDD

RAM:
- Very fast access
- Volatile (loses data)
- Expensive per GB

HDD:
- Slower access
- Non-volatile (keeps data)
- Cheap, large capacity

FDE CYCLE (4-6 marks)

Q: Explain the FDE cycle

1. FETCH:
CPU gets instruction from memory

2. DECODE:
CPU works out what it means

3. EXECUTE:
CPU carries out the instruction

NETWORK QUESTIONS

TOPOLOGIES (4-6 marks)

Q: Compare Bus vs Star topology

BUS:
✓ Cheap to set up
✗ If main cable fails, all fail
✗ Performance degrades

STAR:
✓ More reliable
✓ Easy to add devices
✗ More expensive

SECURITY (3-5 marks)

Q: Match threats to countermeasures

VIRUS → Antivirus software
HACKING → Strong passwords
DATA THEFT → Encryption
UNAUTHORIZED ACCESS → Firewall

Always match security to specific threat!

ALGORITHM DESIGN

PROBLEM SOLVING (8-10 marks)

Q: Write an algorithm to find the largest number in a list

```
# Algorithm: Find largest number
numbers = [5, 12, 3, 9, 7]
largest = numbers[0]
for num in numbers:
  if num > largest:
    largest = num
```

Uses: Sequence (steps in order) + Selection (if) + Iteration (for loop)

EXAM TECHNIQUE GUIDE

QUESTION BREAKDOWN

Code Analysis (4-6 marks):
Trace variables, predict output, find errors

System Components (6-8 marks):
Compare storage, explain FDE, classify hardware

Network Design (4-6 marks):
Choose topology, security measures, performance

HOW TO ANSWER

Code Trace:
Show variable values step by step

Compare Questions:
Give advantages AND disadvantages

Explain Questions:
Use technical terms, give examples

MARK ALLOCATION

- 1 mark = Simple fact/definition
- 2 marks = Comparison with example
- 3+ marks = Detailed explanation
- 6+ marks = Full analysis/algorithm

Always check marks = depth needed!

COMMAND WORDS

State/Name:
Simple answer

Explain:
Say why/how

Compare:
Similarities + differences

SUCCESS STRATEGIES

Programming: ✓ Trace code line by line ✓ Show variable values ✓ Test with different inputs

Systems: ✓ Use FDE terms ✓ Compare storage by speed/cost/capacity ✓ Know different storage types

Networks: ✓ Match security to threats ✓ Compare topology pros/cons ✓ Know LAN vs WAN

General: Read questions twice • Use technical vocabulary • Show all working • Check marks match answer length

GOLDEN RULE: Practice past papers regularly and time yourself!

Remember: Quality over quantity - better to answer fewer questions well than many questions poorly

Key Terms Glossary

Algorithm: Step-by-step instructions to solve a problem

Binary: Number system using only 0s and 1s

Bug: Error in a program

Cache: Very fast temporary storage in CPU

Encryption: Converting data into secret code

Firewall: Security system that monitors network traffic

Loop: Repeated execution of code

Malware: Malicious software designed to harm computers

Protocol: Rules for communication between devices

Variable: Named storage location for data

Exam Tips

✓ **Always show your working** for programming questions ✓ **Use correct technical terminology** from the glossary ✓ **Read questions carefully** - look for command words (explain, compare, evaluate) ✓ **Practice tracing code** - work through programs step by step ✓ **Learn the FDE cycle** - this comes up frequently ✓ **Know storage types** - speed, capacity, cost, volatility ✓ **Understand security threats** and appropriate countermeasures

Remember: Quality of written communication matters - use clear, technical language and structure your answers logically.