



**UTM**  
UNIVERSITI TEKNOLOGI MALAYSIA

Sekolah Pendidikan Profesional dan  
Pendidikan Berterusan  
(UTMSPACE)

---

**FINAL EXAMINATION / PEPERIKSAAN AKHIR**  
**SEMESTER 1 – SESSION 2017 / 2018**  
**PROGRAM KERJASAMA**

SUBJECT CODE : DDPC 2223 / DDC 2223  
KOD KURSUS

COURSE NAME : SYSTEM SOFTWARE /  
NAMA KURSUS PERISIAN SISTEM

YEAR / PROGRAMME : 3 DDPC  
TAHUN / PROGRAM

DURATION : 2 HOURS / 2 JAM  
TEMPOH

DATE : OKTOBER / NOVEMBER 2017  
TARIKH

---

**INSTRUCTION / ARAHAN :**

1. Answer **ALL** questions in SECTION A, B and C in the space provided.  
*Jawab **SEMUA** soalan dalam BAHAGIAN A, B dan C di dalam ruang yang disediakan.*
2. Candidates are required to follow all instructions given out by the examination invigilators.  
*Calon dikehendaki mematuhi semua arahan dari penyelia peperiksaan.*

(You are required to write your name and your lecturer's name on your answer script)  
(Pelajar dikehendaki tuliskan nama dan nama pensyarah pada skrip jawapan )

NAME / NAMA PELAJAR	:	.....
I.C NO./ NO. K/PENGENALAN	:	.....
YEAR / COURSE TAHUN / KURSUS	:	.....
COLLEGE NAME NAMA KOLEJ	:	.....
LECTURER'S NAME NAMA PENSYARAH	:	.....

---

This examination paper consists of ...21.... pages including the cover.  
*Kertas soalan ini mengandungi.. .21....muka surat termasuk kulit hadapan.*



## PUSAT PROGRAM KERJASAMA

### PETIKAN DARIPADA PERATURAN AKADEMIK ARAHAN AM - PENYELEWENGAN AKADEMIK

#### 1. SALAH LAKU SEMASA PEPERIKSAAN

1.1 Pelajar tidak boleh melakukan mana-mana salah laku peperiksaan seperti berikut :-

- 1.1.1 memberi dan/atau menerima dan/atau memiliki sebarang maklumat dalam bentuk elektronik, bercetak atau apa jua bentuk lain yang tidak dibenarkan semasa berlangsungnya peperiksaan sama ada di dalam atau di luar Dewan Peperiksaan melainkan dengan kebenaran Ketua Pengawas; atau
- 1.1.2 menggunakan makluman yang diperolehi seperti di atas bagi tujuan menjawab soalan peperiksaan; atau
- 1.1.3 menipu atau cuba untuk menipu atau berkelakuan mengikut cara yang boleh ditafsirkan sebagai menipu semasa berlangsungnya peperiksaan; atau
- 1.1.4 lain-lain salah laku yang ditetapkan oleh Universiti (seperti membuat bising, mengganggu pelajar lain, mengganggu Pengawas menjalankan tugasnya).

#### 2. HUKUMAN SALAH LAKU PEPERIKSAAN

2.1 Sekiranya pelajar didapati telah melakukan pelanggaran mana-mana peraturan peperiksaan ini, setelah diperakukan oleh Jawatankuasa Peperiksaan Fakulti dan disabitkan kesalahannya, Senat boleh mengambil tindakan dari mana-mana satu yang berikut :-

- 2.1.1 memberi markah SIFAR (0) bagi keseluruhan keputusan peperiksaan kursus yang berkenaan (termasuk kerja kursus); atau
- 2.1.2 memberi markah SIFAR (0) bagi semua kursus yang didaftarkan pada semester tersebut.

2.2 Jawatankuasa Akademik Fakulti boleh mencadangkan untuk diambil tindakan tatatertib mengikut peruntukan Akta Universiti dan Kolej Universiti, 1971, Kaedah-kaedah Universiti Teknologi Malaysia (Tatatertib Pelajar-pelajar), 1999 bergantung kepada tahap kesalahan yang dilakukan oleh pelajar.

2.3 Pelajar yang didapati melakukan kesalahan kali kedua akan diambil tindakan seperti di perkara 2.1.2 dan dicadang untuk diambil tindakan tatatertib mengikut peruntukan Akta Universiti dan Kolej Universiti, 1971, Kaedah-kaedah Universiti Teknologi Malaysia (Tatatertib Pelajar-pelajar), 1999.

**SECTION A TRUE/FALSE QUESTIONS( 10 Marks)**

**Instructions: Answer all questions on page 17.**

**BAHAGIAN A SOALAN BENAR/SALAH (10 Markah)**

**Arahan: Jawab semua soalan pada mukasurat 17.**

1. TRAP2 is an assembler directive.  
*TRAP2 ialah satu direktif penghimpun.*
2. System software acts as the interfaces between the user, the application and the computer.  
*Perisian sistem beraksi sebagai antara muka antara pengguna, aplikasi dan komputer.*
3. Windows CE.NET and Linux OS are examples of embedded operating system.  
*Windows CE.NET dan Linux OS adalah contoh sistem pengoperasian terbenam.*
4. Running program with more than one processor is called multiprogramming.  
*Melaksana program dengan lebih dari satu pemproses dipanggil berbilang pengaturcaraan.*
5. Windows 8.0 is an example of Operating System using GUI interface.  
*Windows 8.0 adalah satu contoh Sistem Pengoperasian mengguna antara muka GUI.*
6. The instruction MOVE 5,D1 will assign the first most significant word of register D1 with value 5.  
*Suruhan MOVE 5,D1 akan mengumpuk perkataan pertama yang paling bererti bagi daftar D1 dengan nilai 5.*
7. The assembler does translates assembler directives into machine codes.  
*Penghimpun menterjemah direktif penghimpun ke dalam kod mesin.*
8. Symbol Table is a table that stores address of all labels or symbols used in a source program.  
*Jadual Simbol adalah satu jadual yang menyimpan alamat semua label atau simbol dalam satu program sumber.*
9. XXX DC.B %00001111 cause the label XXX to not be assigned with value \$00000F.  
*XXX DC.B %00001111 menyebabkan label XXX tidak diberi nilai \$00000F.*
10. MOVE.L #@77,A1 has an error.  
*MOVE.L #@77,A1 mempunyai ralat.*



**SECTION B      OBJECTIVE QUESTIONS (15 Marks)**

**Instructions:** Choose the most appropriate answer.

**Answer all questions on page 17.**

**BAHAGIAN B      SOALAN OBJEKTIF (15 Markah)**

**Arahan:** *Pilih Jawapan yang paling tepat.*

**Jawab semua soalan pada muka surat 17.**

1.     **Parsing** is also known as?  
*Imbasan juga dikenali sebagai?*
  - A. Lexical Analysis / *Analisis Leksikal*
  - B. Semantic Analysis / *Analisis Semantik*
  - C. Syntax Analysis / *Analisis Sintak*
  - D. None of the above / *Tiada di atas*
  
2.     The **Linker** \_\_\_\_\_.  
**Pemaut** \_\_\_\_\_.
  - A. is the same as a loader / *adalah sama dengan pemuat*
  - B. is used before program is executed / *digunakan sebelum program*
  - C. is the same as an editor / *adalah sama dengan penyunting*
  - D. None of the above / *Tiada di atas*

DDPC 2223 / DDC 2223

3. In a **two pass assembler** the symbol table generation occurs during  
*Dalam satu **penghimpun dua laluan** penjanaan jadual simbol berlaku semasa*
- A. the second pass / *laluan kedua*
  - B. the first pass / *laluan pertama*
  - C. the third pass / *laluan ketiga*
  - D. None of the above / *Tiada di atas*
4. Which of the following cannot be done by an **assembler**?  
*Manakah antara berikut tidak boleh dilakukan oleh satu **penghimpun**?*
- A. Convert Assembler to Machine Code / *Tukar Penghimpun kepada Kod Mesin*
  - B. Generate Symbol Table / *Jana Jadual Simbol*
  - C. Generate Program Listing / *Jana Senarai Program*
  - D. All of the above / *Semua di atas*
5. Which of the following is **not** a part of the phases in **compiling process**?  
*Manakah antara berikut **bukan** sebahagian fasa dalam **proses pengkompilan**?*
- A. Lexical Analysis / *Analisis Leksikal*
  - B. Semantic Analysis / *Analisis Semantik*
  - C. Code Analysis / *Analisis Kod*
  - D. None of the above / *Tiada di atas*



DDPC 2223 / DDC 2223

9. The following Operating System is embedded **EXCEPT:**  
*Sistem Pengoperasian berikut adalah*
- A. Windows CE.NET      C. Windows Mobile 2003  
B. Windows XP      D. Palm OS
7. The following UNIX command are correctly used **EXCEPT:**  
*Perintah UNIX berikut digunakan dengan betul **KECUALI:***
- A. \$chmod 333 txt.c      C. \$man man  
B. \$cd ..      D. \$whoami

**SECTION C      STRUCTURED QUESTIONS (75 Marks)**

**Instructions:** Answer all questions in the space provided.

**BAHAGIAN      SOALAN BERSTRUKTUR (75 Markah)**

**Arahan:** Jawab semua soalan pada ruang yang disediakan.

1. a) Give **three(3)** example of **Utility Software**.

[3M]

*Berikan **tiga(3)** contoh **Perisian Utiliti**.*

b) Describe the following characteristics of **Operating System**:

[6M]

*Terangkan ciri-ciri **Sistem Pengoperasian** berikut:*

i) *Realtime / Masa Nyata*

ii) *Batch Processing / Pemprosesan Kelompok*

iii) *Timesharing / Kongsi Masa*



DDPC 2223 / DDC 2223

2. With reference to the following data registers, address registers and main memory below, show the content of the register or main memory location that has changed, after the execution of the following instruction. [10M]

*Merujuk kepada daftar data, daftar alamat dan ingatan utama di bawah, tunjukkan kandungan daftar atau lokasi ingatan utama yang telah bertukar, setelah pelaksanaan arahan berikut.*

DATA REGISTER

D0	1111 1111
D1	2222 2222
D2	0001 0010
D3	4477 FFF4

ADDRESS REGISTER

A0	0022 0300
A1	0022 0302
A2	0022 0304
A3	0022 0306

MAIN MEMORY

\$220300	AA	88
\$220302	BB	99
\$220304	FF	AD
\$220306	33	BD

- a) CLR.W (A2)

D0	1111 1111
D1	2222 2222
D2	0001 0010
D3	4477 FFF4

A0	0022 0300
A1	0022 0302
A2	0022 0304
A3	0022 0306

\$220300		
\$220302		
\$220304		
\$220306		

- b) SUB.W D1,(A2)+

D0	1111 1111
D1	2222 2222
D2	0001 0010
D3	4477 FFF4

A0	0022 0300
A1	0022 0302
A2	
A3	0022 0306

\$220300		
\$220302		
\$220304		
\$220306		

- c) ADD.W #\$7777,2(A0)

D0	1111 1111
D1	2222 2222
D2	0001 0010
D3	4477 FFF4

A0	0022 0300
A1	0022 0302
A2	0022 0304
A3	0022 0306

\$220300		
\$220302		
\$220304		
\$220306		

DDPC 2223 / DDC 2223

d) MULS D3,D2

D0	1111 1111
D1	2222 2222
D2	
D3	4477 FFF4

A0	0022 0300
A1	0022 0302
A2	0022 0304
A3	0022 0306

\$220300		
\$220302		
\$220304		
\$220306		

e) DIVS #5,D2

D0	1111 1111
D1	2222 2222
D2	
D3	4477 FFF4

A0	0022 0300
A1	0022 0302
A2	0022 0304
A3	0022 0306

\$220300		
\$220302		
\$220304		
\$220306		

3. a) Why are **Mnemonic code** used? [2M]  
*Mengapakah **Kod Mneumonik** digunakan?*

- b) Give five(5) examples of **Arithmetic Instruction** for **Motorola 68000** microprocessor. [5M]  
*Berikan five(3) contoh **Arahan Aritmetik** bagi mikro pemproses **Motorola 68000**.*

- c) Draw the diagram how **Editor, Assembler, Compiler, Linker** and **Loader** functions with respect to **Source Code, Object Code** and **Executable Code**. [5M]  
*Lukiskan gambar rajah bagaimana **Penyunting, Penghimpun, Pengkompil, Pemuat** dan **Pemuat** berfungsi berdasarkan kepada **Kod Sumber, Kod Objek** dan **Kod BolehLaksana**.*

- d) What is the disadvantage of **One Pass Assembler**?  
*Apakah kekurangan **Penghimpun Satu Laluan**?*

[2M]

4. With reference to **APPENDIX B**, assemble the source code below manually and built it's machine code.

[16M]

*Merujuk kepada **APPENDIX B**, himpunkan kod sumber di bawah secara manual dan bina kod mesinnya.*

ADDRESS ALAMAT	MACHINE CODE KOD MESIN	SOURCE CODE KOD SUMBER
00001000		1 ORG \$1000
00001000		2 START:
00001000		3 ADD.B X,D0
00001006		4 ADD.L Y,D1
0000100C		5 ADD D5,D7
0000100E		6 ADD.B Y,D0
00001014		7 ADD.L #255,D0
00001018		8 ADD D7,D6
0000101A		9 ADD.B D0,Z
00000700		10 ORG \$700
00000700	=	11 Y: DC.B 10
00000701	=	12 X: DC.B 100
00000702		13 Z: DS.B 1

5. Given the PASCAL statement  $X := A \text{ DIV } C + B * 60$ , state the function of each stage and its generated product during compilation. Refer to **APPENDIX A**. [15M]

*Diberi pernyataan PASCAL  $X := A \text{ DIV } C + B * 60$ , nyatakan fungsi setiap tahap dan hasil yang dijana semasa pengkompilan. Rujuk pada **APPENDIX A**.*

a) Lexical Analysis / Analisis Leksikal

b) Syntax Analysis / Analisis Sintak



c) Code Generation / *Penjanaan Kod*

6. Write a 68K ASSEMBLY CODE at the starting address \$200 for the arithmetic expression  $Y = A * B - C \text{ DIV } D + E$  using SUB, ADD, DIVS, MULS and MOVE instructions. [11M]

*Tuliskan satu KOD PENGHIMPUN 68K pada alamat permulaan \$200 untuk pernyataan aritmetik  $Y = A * B - C \text{ DIV } D + E$  menggunakan arahan SUB, ADD, DIVS, MULS.*

```
ORG $1000
A DC.W 4
B DC.W -14
C DC.W 114
D DC.W 24
E DC.W 4
Y DS.W 2
```

```
ORG $200
```



**ANSWER SHEET FOR SECTION A AND SECTION B**  
**RUANG JAWAPAN BAHAGIAN A DAN BAHAGIAN B**

---

**SECTION A (10 Marks)**  
**BAHAGIAN A (10 Markah)**


1		6	
2		7	
3		8	
4		9	
5		10	

**SECTION B (15 Marks)**  
**BAHAGIAN B (15 Markah)**

1		6	
2		7	
3		8	
4		9	
5		10	

APPENDIX A

PASCAL LANGUAGE SYNTAX / *SINTAK BAHASA PASCAL*

1. `<prog> ::= PROGRAM <prog-name> VAR <dec-list> BEGIN  
    <stmt-list> END`
2. `<prog-name> ::= id`
3. `<dec-list> ::= <dec> | <dec> `



APPENDIX B / LAMPIRAN B  
MC60000 INSTRUCTION FORMAT / FORMAT ARAHAN MC60000

Integer Instructions

**ADD**

Add  
(M68000 Family)

**ADD**



APPENDIX B / LAMPIRAN B  
MC60000 INSTRUCTION FORMAT / FORMAT ARAHAN MC60000

Integer Instructions

**ADD**

Add  
(M68000 Family)

**ADD**

Instruction Fields:

Register field—Specifies any of the eight data registers.

Opmode field

Byte	Word	Long	Operation
000	001	010	< ea > + Dn → Dn
100	101	110	Dn + < ea > → < ea >

Effective Address field—Determines addressing mode.

- a. If the location specified is a source operand, all addressing modes can be used as listed in the following tables:

Addressing Mode	Mode	Register
Dn	000	reg. number:Dn
An*	001	reg. number:An
(An)	010	reg. number:An
(An) +	011	reg. number:An
-(An)	100	reg. number:An
(d <sub>16</sub> ,An)	101	reg. number:An
(d <sub>8</sub> ,An,Xn)	110	reg. number:An

Addressing Mode	Mode	Register
(xxx).W	111	000
(xxx).L	111	001
#<data>	111	100
(d <sub>16</sub> ,PC)	111	010
(d <sub>8</sub> ,PC,Xn)	111	011

MC68020, MC68030, and MC68040 only

(bd,An,Xn)**	110	reg. number:An
([bd,An,Xn],od)	110	reg. number:An
([bd,An],Xn,od)	110	reg. number:An

(bd,PC,Xn)†	111	011
([bd,PC,Xn],od)	111	011
([bd,PC],Xn,od)	111	011

\*Word and long only

\*\*Can be used with CPU32.

APPENDIX B / LAMPIRAN B

MC60000 INSTRUCTION FORMAT / FORMAT ARAHAN MC60000

Integer Instructions

**ADD**

**Add**  
(M68000 Family)

**ADD**

- b. If the location specified is a destination operand, only memory alterable addressing modes can be used as listed in the following tables:

Addressing Mode	Mode	Register
Dn	—	—
An	—	—
(An)	010	reg. number:An
(An) +	011	reg. number:An
-(An)	100	reg. number:An
(d <sub>16</sub> ,An)	101	reg. number:An
(d <sub>8</sub> ,An,Xn)	110	reg. number:An

Addressing Mode	Mode	Register
(xxx).W	111	000
(xxx).L	111	001
#<data>	—	—
(d <sub>16</sub> ,PC)	—	—
(d <sub>8</sub> ,PC,Xn)	—	—

MC68020, MC68030, and MC68040 only

(bd,An,Xn)*	110	reg. number:An
((bd,An,Xn),od)	110	reg. number:An
((bd,An),Xn,od)	110	reg. number:An

(bd,PC,Xn)*	—	—
((bd,PC,Xn),od)	—	—
((bd,PC),Xn,od)	—	—

\*Can be used with CPU32

**NOTE**

The Dn mode is used when the destination is a data register; the destination < ea > mode is invalid for a data register.

ADDA is used when the destination is an address register. ADDI and ADDQ are used when the source is immediate data. Most assemblers automatically make this distinction.

**Mukasurat ini sengaja dibiarkan kosong**

*[ This page is purposely left blank ]*