Cover Page

Information

Department of Computer Science

Examination paper for TDT4258 Low Level Programming

Examination date: 18-January-2022

Examination time (from-to): 09:00 to 12:00

Permitted examination support material: A / All support material is allowed

If you experience technical problems during the exam, contact Orakel support services as soon as possible before the examination time expires/the test closes. If you don't get through immediately, hold the line until your call is answered.

OTHER INFORMATION

Do not open Inspera in multiple tabs, or log in on multiple devices, simultaneously. This may lead to errors in saving/submitting your answer.

Get an overview of the question set before you start answering the questions.

Read the questions carefully, make your own assumptions and specify them in your answer. Only contact academic contact if you think there are errors or insufficiencies in the question set.

Cheating/Plagiarism: The exam is an individual, independent work. Examination aids are permitted, but make sure you follow any instructions regarding citations. During the exam it is not permitted to communicate with others about the exam questions or distribute drafts for solutions. Such communication is regarded as cheating. All submitted answers will be subject to plagiarism control. *Read more about cheating and plagiarism here.*

Notifications: If there is a need to send a message to the candidates during the exam (e.g. if there is an error in the question set), this will be done by sending a notification in Inspera. A dialogue box will appear. You can re-read the notification by clicking the bell icon in the top right-hand corner of the screen. All candidates will also receive an SMS to ensure that nobody misses out on important information. Please keep your phone available during the exam.

Weighting: There are 10 questions in the exam and each question is worth 5 points.

ABOUT SUBMISSION

Answering in Inspera: If the question set contains questions that are not upload assignment, you must answer them directly in Inspera. In Inspera, your answers are saved automatically every 15 seconds.

NB! We advise against pasting content from other programs, as this may cause loss of formatting and/or entire elements (e.g. images, tables).

Automatic submission: Your answer will be submitted automatically when the examination time expires and the test closes, as long as you have answered at least one question. This will happen even if you do not click "Submit and return to dashboard" on the last page of the question set. You can reopen and edit your answer as long as the test is open. If no questions are answered by the time the examination time expires, your answer will not be submitted. This is considered as "did not attend the exam".

Withdrawing from the exam: If you become ill during the exam or wish to submit a blank answer/withdraw from the exam for another reason, go to the menu in the top right-hand corner and click "Submit blank". This cannot be undone, even if the test is still open.

Accessing your answer post-submission: You will find your answer in Archive when the examination time has expired.

1 Embedded Systems

List five differences between embedded systems and general purpose computing systems.
(Max. Marks: 5)

Fill in your answer here

Format | Σ

Words: 0

Maximum marks: 5

2 ARM Assembly Instructions

In ARM assembly programming, a "mov" instruction can be used to move an immediate value, say 45, to a register, say r3, as below:

mov r3, #45.

Can we move a value of 1024 to a register using this same instruction? **(Max. Marks: 2)**
Provide a justification for your answer. **(Max. Marks: 3)**

Fill in your answer here

Format | Σ

Words: 0

Maximum marks: 5

1

3 ARM Assembly Programming

Consider the following ARM assembly code:

```
push   r1
push   r2
str    r3, [sp, #4]
pop    r1
pop    r2
```

The register values before executing this code are as follows:

```
r1:   0x40
r2:   0x80
r3:   0x20
sp:   0x4000
```

Describe how each instruction in the instruction sequence given above will modify the register and/or memory values. **(Max. Marks: 5)**

Notice that, in your answer, you need to write the new register and/or memory values after executing every single instruction. If the value of a register and/or a memory location does not change, you do not need to include it in your answer.

Fill in your answer here

Format

Words: 0

Maximum marks: 5

4 I/O Device Communication

"Polling" and "Interrupt" are the two mechanisms that can be used to check the readiness of input/output devices for communication/data-transfer. What is the difference between these two mechanisms? **(Max. Marks: 2)**

List one advantage and one disadvantage of each of these two mechanisms. **(Max. Marks: 2)**

List the steps taken when an interrupt arrives until the interrupt handling is complete. **(Max. Marks: 1)**

Fill in your answer here

Format | Σ

Words: 0

Maximum marks: 5

1

5 Caches

How many bits are needed for **Tag, Index, and Byte Offset** in an 32KB 16-way set associative cache with 64-byte block size, assuming 32-bit addresses? Please elaborate how you calculated the numbers of bits. **(Max. Marks: 5)**

Note: Cache size refers only to the size of data portion, not including tags and valid bits. So, a 32KB cache can store 32KB of data.

Fill in your answer here

Format

Words: 0

Maximum marks: 5

6 Virtual Memory

Consider a system with 32-bit virtual addresses and 28-bit physical addresses. Due to the limited physical memory, the designers want to keep the page table smaller than 1MB. What is the smallest page size that allows to keep the page table size smaller than 1MB? Please elaborate how you determined this page size. **(Max. Marks: 5)**

For your calculations, assume a single level page table with each entry containing 6 status bits and a physical page number. Also assume that both physical and virtual pages are same size.

Fill in your answer here

Format

Words: 0

Maximum marks: 5

1

7 C Programming

Consider that the following C structure is used to keep employee information at an organization:

```
struct employee {
      char name[100];
      int salary;
      uint8_t status;
}
```

The structure stores information about and employee's name, salary, and status. The individual bits in the "status" variable provide the following information:

						is_permanent	is_manager	is_onLeave

is_onLeave (bit 0): This bit is "1" if the employee is currently on leave, otherwise it is 0.
is_manager (bit 1): This bit is "1" if the employee is a manager, otherwise it is 0.
is_permanent (bit 2): This bit is "1" if the employee is a permanent employee, otherwise 0.
The rest of the bits in the "status" variable are currently unused.

Your Task:

The organization has decided that the bit 3 of the "status" variable should be used as is_highEarner. You need to finish the implementation of *set_high_earner* function below to set the is_highEarner bit in "status" variable **if the employee is a manager and has salary more than 800000**.

```
void set_high_earner() {
      struct employee elist[200];

      /*Assume that some code here initializes the elist array to fill name, salary, and status
        (but does not fill the is_highEarner bit of "status")*/

     for (int i = 0; i < 200; i++) {
         //Your code for setting the is_highEarner bit goes here
     }
}
```

Please comment your code appropriately.

Fill in your answer here

```
1
```

Maximum marks: 5

8 Compilers

a) Apply the following three optimizations in following order:

1. Dead code elimination
2. Constant propagation
3. Constant folding

to the following piece of C code repeatedly until no more optimizations are possible. Show the resulting code and indicate the modified and/or eliminated operations **after each optimization step**. **(Max. Marks: 3)**

```
int x = 30;
int y = x / 5;
y = 9 - y;
int z;
z = y * 4;
if (z > 10) {
    z = z - 10;
} else {
    z = z + 10;
}
return z * (60 / x);
```

b) Explain why it is necessary to apply these optimizations repeatedly, instead of just once, to fully optimize the code. **(Max. Marks: 2)**

Fill in your answer here

Format | Σ

Words: 0

Maximum marks: 5

9 Operating Systems

a) Why cannot round-robin scheduling guarantee that a process will finish within a given time, .i.e. before deadline? **(Max. marks: 3)**

b) A process can be in one of the following states: running, ready, and blocked. What events can cause the following state transitions (assume preemptive scheduling.)? **(Max. marks: 2)**

1) running to blocked
2) running to ready

Fill in your answer here

Format

Words: 0

Maximum marks: 5

10 Power in Digital Circuits

Define the terms "static power consumption" and "dynamic power consumption". **(Max. Marks: 1)**
What causes static and dynamic power consumption in digital circuits? **(Max. Marks: 2)**
Why is the dynamic power consumption in CMOS circuits very low? **(Max. Marks: 2)**

Fill in your answer here

Format

Σ

Words: 0

Maximum marks: 5